Susan Chipman

The Twelfth Annual Conference of the Cognitive Science Society

July 25 - 28, 1990

Cambridge, Massachusetts



The publication of these proceedings has been made possible by the generosity of Olivetti Research (Ivrea, Italy)

Program of the

Twelfth Annual Conference of the Cognitive Science Society

25-28 July 1990
Cambridge, Massachusetts

LEA LAWRENCE ERLBAUM ASSOCIATES, PUBLISHERS
1990 Hillsdale, New Jersey Hove and London

Copyright 1990 by the Cognitive Science Society.

All rights reserved. No part of this book may be reproduced in any form, by photostat, microform, retrieval system, or by any other means, without the prior written permission of the publisher.

Distributed by

Lawrence Erlbaum Associates, Inc.
365 Broadway
Hillsdale, New Jersey 07642

ISBN 0-8058-0938-4

ISSN 1047-1316

Printed in the United States of America

The Twelfth Annual Conference of the Cognitive Science Society

July 25-28, 1990

Cambridge, Massachusetts

Conference Chair: Massimo Piattelli-Palmarini (MIT)

Scientific Advisors: Beth Adelson (Tufts), Stephen M. Kosslyn (Harvard), Steven Pinker (MIT), Kenneth Wexler (MIT)

Reviewers

Richard Alterman
William Bechtel
Thomas Bever
Lawrence Birnbaum
Paul Bloom
Michael Brent
Yves Chauvin
Michelene Chi
Garrison Cottrell
Stephen Crain
Bonnie Dorr
Kevin Dunbar
Michael Dyer
Jeffrey Elman
Nancy Etcoff
Brian Falkenhainer
Douglas Fisher
Margot Flowers
Kenneth Forbus
David Forster
Deborah Frisch
Dedre Gentner
Mark Gluck
Robert Goldstone
Arthur Graesser
Ken Haase
Kristian Hammond
Gilbert Harman
Keith Holyoak
Michael Jordan
John Kim
Eric Lormand
Maryellen MacDonald
Charles Martin
Robert Mathews
Douglas Medin
Scott Meredith
Janet Metcalfe
Nancy Nersessian
Allen Newell
Padraig O'Seaghdha
Christopher Owens
Caroline Palmer
Michael Pazzani
Steven Pinker
Peter Pirolli
Alan Prince
Lauren Resnick
Whitman Richards
Eric Ristad
Jay Rueckl
Dirk Ruiz
Mark Seidenberg
Colleen Seifert
Lokendra Shastri
Jeff Shrager
Thomas Shultz
Paul Smolensky
James Spohrer
Robert Stalnaker
Josh Stern
Katia Sycara
Carol Tenny
Paul Thagard
Elise Turner
Roy Turner
Kurt VanLehn
Stella Vosniadou
David Waltz
Kenneth Wexler
Tony Wilkie
Jeremy Wolfe
Roy Zito-Wolf

Governing Board

David Rumelhart, Chair	August 1986 - August 1992
Ann Brown	August 1984 - August 1990
Dedre Gentner	August 1987 - August 1993
James Greeno, Chair Elect	August 1987 - August 1993
Patrick Hayes	August 1984 - August 1990
Geoffrey Hinton	August 1986 - August 1992
Walter Kintsch	August 1988 - August 1994
Steve Kosslyn	August 1989 - August 1995
George Lakoff	August 1989 - August 1995
Wendy Lehnert	August 1989 - August 1995
James L. McClelland	August 1988 - August 1994
Ulric Neisser	August 1985 - August 1991
Donald A. Norman	August 1985 - August 1991
John Anderson, Past Chair	August 1983 - August 1989

Ex Officio Board Members

Martin Ringle, Executive Editor of *Cognitive Science*	August 1986 -
Alan Lesgold, Secretary/Treasurer	August 1988 - August 1991

Proceedings of the
Twelfth Annual Conference of the Cognitive Science Society
Cambridge, Massachusetts - July 25/28, 1990

PAPER PRESENTATIONS

Group I: Reasoning

I.1. Analogy

Effect of Structure of Analogy and Depth of Encoding on Learning Computer Programming 1
Yam San Chee, National University of Singapore

Evaluating and Debugging Analogically Acquired Models 9
Beth Adelson, Tufts University

Analogical Process Performance 16
Clark N. Quinn, University of Pittsburgh

The Effects of Familiar Labels on Young Children's Performance in an Analogical Mapping Task 22
Mary Jo Rattermann, University of Illinois
Dedre Gentner, University of Illinois
Judy DeLoache, University of Illinois

Representational Issues in Analogical Transfer 30
Colleen M. Seifert, University of Michigan
Kenneth C. Gray, University of Michigan

Analogical Mapping During Similarity Judgments 38
Arthur B. Markman, University of Illinois
Dedre Gentner, University of Illinois

Goal Similarity in Analogical Problem Solving 45
Bruce D. Burns, University of California, Los Angeles

Internal Analogy: A Model of Transfer within Problems 53
Angela Kennedy Hickman, Carnegie Mellon University
Jill H. Larkin, Carnegie Mellon University

Making SME Greedy and Pragmatic 61
Kenneth D. Forbus, Beckman Institute, University of Illinois
Dan Oblinger, Beckman Institute, University of Illinois

Analogical Interpretation in Context 69
Brian Falkenhainer, Palo Alto Research Center

I.2. Case-Based Reasoning
Multiple Abstracted Representations in Problem Solving and Discovery in Physics 77
Nancy J. Nersessian, Princeton University
James G. Greeno, Stanford University/Institute for Research on Learning

Viewing Design as a Cooperative Task 85
Katia P. Sycara, Carnegie Mellon University

The Temporal Nature of Scientific Discovery: The Roles of Priming and Analogy 93
Kevin Dunbar, McGill University
Christian D. Schunn, McGill University

Reasoning Directly from Cases in a Case-Based Planner 101
Robert McCartney, University of Connecticut

An Internal Contradiction of Case-Based Reasoning 109
David B. Skalak, University of Massachusetts

Qualitative Reasoning about the Geometry of Fluid Flow 117
Hyun-Kyung Kim, Beckman Institute, University of Illinois

I. 3. Categories and Concepts
Is There a Default Similarity Distance for Categories? 125
Yaakov Kareev, Hebrew University of Jerusalem
Judith Avrahami, Hebrew University of Jerusalem

Judgment, Graininess, and Categories 133
Ilan Yaniv, University of Chicago
Dean Foster, University of Chicago

Learning Attribute Relevance in Context in Instance-Based Learning Algorithms 141
David W. Aha, University of California, Irvine
Robert L. Goldstone, University of Michigan

Effects of Background Knowledge on Family Resemblance Sorting 149
Woo-Kyoung Ahn, University of Michigan

Superordinate and Basic Level Categories in Discourse: Memory and Context 157
Sandra L. Peters, State University of New York at Buffalo
William J. Rapaport, State University of New York at Buffalo

Learning Overlapping Categories 166
Joel D. Martin, Georgia Institute of Technology

The Right Concept at the Right Time: How Concepts Emerge as Relevant in Response to Context-Dependent Pressures 174
Melanie Mitchell, Indiana University
Douglas R. Hofstadter, Indiana University

Classification of Dot Patterns with Competitive Chunking 182
Emile Servan-Schreiber, Carnegie Mellon University

I.4. Causality, Induction and Decision-Making

Can Causal Induction Be Reduced to Associative Learning? 190
Michael R. Waldmann, University of Frankfurt
Keith J. Holyoak, University of California, Los Angeles

Decision Models: A Theory of Volitional Explanation 198
Ashwin Ram, Georgia Institute of Technology

Knowledge Goals: A Theory of Interestingness 206
Ashwin Ram, Georgia Institute of Technology

The Dempster-Shafer Theory of Evidence as a Model of Human Decision Making 215
Donald H. Mitchell, Amoco Production Company

Feature Selection and Hypothesis Selection Models of Induction 221
Michael J. Pazzani, University of California, Irvine
Glenn Silverstein, University of California, Irvine

A Rule Based Model of Judging Harm-Doing 229
Thomas R. Shultz, McGill University

I.5. Planning
The Mechanism of Restructuring in Geometry 237
Stellan Ohlsson, University of Pittsburgh

Noticing Opportunities in a Rich Environment 245
Matthew Brand, Northwestern University
Lawrence Birnbaum, Northwestern University

Recognizing Novel Uses for Familiar Plans 253
Beth Adelson, Tufts University

Planning to Learn 261
Lawrence Hunter, National Library of Medicine

Brainstormer: A Model of Advice-Taking 269
Eric K. Jones, Northwestern University

Representing Abstract Plan Failures 277
Christopher Owens, University of Chicago

I.6. Rule-Based Reasoning, Explanation and Problem-Solving
Reasoning with Function Symbols in a Connectionist System 285
Venkat Ajjanagadde, University of Pennsylvania

Not all Potential Cheaters are Equal: Pragmatic Strategies in Deductive Reasoning 293
Vittorio Girotto, C.N.R., Rome
Paolo Legrenzi, University of Trieste

Explanations in Cooperative Problem Solving Systems 301
Thomas Mastaglio, University of Colorado, Boulder
Brent Reeves, University of Colorado, Boulder

Improving Explanatory Competence 309
Kenneth S. Murray, University of Texas, Austin

Incremental Envisioning: The Flexible Use of Multiple Representations in Complex Problem Solving 317
Malcolm I. Bauer, Princeton University
Brian J. Reiser, Princeton University

Task-Based Criteria for Judging Explanations 325
David B. Leake, Indiana University

Are There Developmental Milestones in Scientific Reasoning? 333
Anne L. Fay, Carnegie Mellon University
David Klahr, Carnegie Mellon University
Kevin Dunbar, McGill University

Why Fodor and Pylyshyn Were Wrong: The Simplest Refutation 340
David J. Chalmers, Indiana University

Group II: Language

II.1. Phonology

Phonological Rule Induction: An Architectural Solution 348
David S. Touretzky, Carnegie Mellon University
Gillette Elvgren III, Carnegie Mellon University
Deirdre W. Wheeler, University of Pittsburgh

Discovering Faithful "Wickelfeature" Representations in a Connectionist Network 356
Michael C. Mozer, University of Colorado, Boulder

Constraints on Assimilation in Vowel Harmony Languages 364
Mary Hare, University of California, San Diego

II.2. Syntax

** Recency Preference and Garden-Path Effects 372
Edward Gibson *(Recipient of the David Marr Prize for 1990)*
Carnegie Mellon University

A Connectionist Treatment of Grammar for Generation 380
Nigel Ward, University of California, Berkeley

Harmonic Grammar - A Formal Multi-Level Connectionist Theory of Linguistic Well-Formedness: Theoretical Foundations 388
Géraldine Legendre, University of Colorado, Boulder
Yoshiro Miyata, University of Colorado, Boulder
Paul Smolensky, University of Colorado, Boulder

A Parallel Constraint Satisfaction and Spreading Activation Model for Resolving Syntactic Ambiguity 396
Suzanne Stevenson, University of Maryland

Functional Constraints on Backwards Pronominal Reference 404
Catherine L. Harris, University of California, San Diego
Elizabeth A. Bates, University of California, San Diego

II.3. Lexical Semantics

A Semantic Analysis of Action Verbs Based on Physical Primitives 412
Jugal K. Kalita, University of Pennsylvania
Norman I. Badler, University of Pennsylvania

Some Principles of the Organization of Verbs in the Mental Lexicon 420
Christiane Fellbaum, Princeton University
Roger Chaffin, Trenton State College

Semantic Classification of Verbs from their Syntactic Contexts: Automated Lexicography with Implications for Child Language Acquisition 428
Michael R. Brent, Massachusetts Institute of Technology

Sense Generation, or How to Make the Mental Lexicon Flexible 438
Bradley Franks, University of Edinburgh
Nick Braisby, University of Edinburgh

A Distributed Feature Map Model of the Lexicon 447
Risto Miikkulainen, University of California, Los Angeles

Efficient Learning of Language Categories: The Closed-Category Relevance Property and Auxiliary Verbs 455
Sheldon Nicholl, University of Illinois, Urbana
David C. Wilkins, University of Illinois, Urbana

The Representation of Word Meaning 463
Renison J. Gonsalves, Brooklyn College

Lexical Cooccurrence Relations in Text Generation 471
Leo Wanner, GMD-IPSI, Darmstadt
John A. Bateman, University of Southern California

Learning Lexical Knowledge in Context: Experiments with Recurrent Feed Forward Networks 479
Steven L. Small, University of Pittsburgh

II.4. Pragmatics and Communication

How to Describe What? Towards a Theory of Modality Utilization 487
Yigal Arens, University of Southern California
Eduard Hovy, University of Southern California

Towards a Failure-Driven Mechanism for Discourse Planning: A Characterization of Learning Impairments 495
Ingrid Zukerman, Monash University

Coherence Relation Reasoning in Persuasive Discourse 503
Horng Jyh P. Wu, University of Michigan
Steven L. Lytinen, University of Michigan

Analyzing Research Papers Using Citation Sentences 511
Wendy Lehnert, University of Massachusetts
Claire Cardie, University of Massachusetts
Ellen Riloff, University of Massachusetts

Participating in Plan-Oriented Dialogs 519
Alex Quilici, University of California, Los Angeles

Fictional Narrative Comprehension: Structuring the Deictic Center 526
Erwin M. Segal, State University of New York at Buffalo

II.5. Language Processing

Thematic Roles and Pronoun Comprehension 534
Rosemary J. Stevenson, University of Durham
Rosalind A. Crawley, University of Durham
Garry Wilson, University of Durham
David Kleinman, University of Durham

Parallel Processes During Question Answering? 542
Scott P. Robertson, Rutgers University
Kimberly C. Weber, Rutgers University

Spreading Activation in PDP Networks 550
James Hendler, University of Maryland

Resolution of Structural Ambiguities in Sentence Comprehension: On-Line Analysis of Syntactic, Lexical, and Semantic Effects 558
Gerhard Strube, Ruhr University
Barbara Hemforth, Ruhr University
Heike Wrobel, Ruhr University

Group III: Vision

III.1. Attention

A Connectionist Model of Attentional Enhancement and Signal Buffering 566
Judith M. Shedden, University of Pittsburgh
Walter Schneider, University of Pittsburgh

Visual Search as Constraint Propagation 574
Peter A. Sandon, Dartmouth College
Berrin A. Yanikoglu, Dartmouth College

A Computer Model of 2D Visual Attention 582
Mark Wiesmeyer, University of Michigan
John Laird, University of Michigan

On the Evolution of a Visual Percept 590
John Theios, University of Wisconsin
Stephen T. Morgan, University of Wisconsin

III.2. Objects and Features

A Computational Model of Visual Pattern Discrimination in Toads 598
DeLiang Wang, University of Southern California
Michael A. Arbib, University of Southern California

Binding and Type-Token Problems in Human Vision 606
Nancy G. Kanwisher, University of California, Berkeley

Dynamic Binding: A Basis for the Representation of Shape by Neural Networks 614
John E. Hummel, University of Minnesota
Irving Biederman, University of Minnesota

Caricature Recognition in a Neural Network 622
James W. Tanaka, Carnegie Mellon University

Equilateral Triangles: A Challenge for Connectionist Vision 629
Subutai Ahmad, International Computer Science Institute, Berkeley
Stephen Omohundro, International Computer Science Institute, Berkeley

The Computation of Emotion in Facial Expression Using the Jepson & Richards Perception Theory 637
Brent C.J. Britton, Massachusetts Institute of Technology

III.3. Imagery

Imagery and Problem Solving 646
Yulin Qin, Carnegie Mellon University
Herbert A. Simon, Carnegie Mellon University

Psychological Simulation and Beyond 654
Ian Pratt, University of Manchester

Formal Models for Imaginal Deduction 662
George W. Furnas, Bell Communications Research

Integrating Imagery and Visual Representations 670
B. Chandrasekaran, Ohio State University
N.H. Narayanan, Ohio State University

Group IV: Learning and Memory

IV. 1. Acquisition

Language Acquisition via Strange Automata 678
Jordan B. Pollack, Ohio State University

Miniature Language Acquisition: A Touchstone for Cognitive Science 686
Jerome A. Feldman, International Computer Science Institute, Berkeley
George Lakoff, International Computer Science Institute, Berkeley
Andreas Stolcke, International Computer Science Institute, Berkeley
Susan Hollbach Weber, International Computer Science Institute, Berkeley

Routine Evolution as the Microgenetic Basis of Skill Acquisition 694
Phil Agre, University of Chicago
Jeff Shrager, Xerox Palo Alto Research Center

The Disruptive Potential of Immediate Feedback 702
Lael J. Schooler, Carnegie Mellon University
John R. Anderson, Carnegie Mellon University

Learning the Structure of Event Sequences 709
Axel Cleeremans, Carnegie Mellon University
James L. McClelland, Carnegie Mellon University

Explanation-Based Learning of Correctness: Towards a Model of the Self-Explanation Effect 717
Kurt VanLehn, Carnegie Mellon University
William Ball, Carnegie Mellon University
Bernadette Kowalski, Carnegie Mellon University

Indexing Libraries of Programming Plans 725
James C. Spohrer, Apple Computer

Rule Induction and Interference in the Absence of Feedback: A Classifier System Model 733
Lewis G. Roussel, Louisiana State University
Robert C. Mathews, Louisiana State University
Barry B. Druhan, Louisiana State University

IV.2. Memory

Associative Memory-Based Reasoning: Some Experimental Results 741
Boicho Nikolov Kokinov, Bulgarian Academy of Sciences

Uniformity of Associative Impairment in Amnesia 750
Gregory V. Jones, University of Warwick
Siobhan B.G. MacAndrew, University of Warwick

Predictive Utility in Case-Based Memory Retrieval 757
Hollyn M. Johnson, University of Michigan
Colleen M. Seifert, University of Michigan

Episodic Memory in Connectionist Networks 764
Chris A. Kortge, Stanford University

On the Domain Specificity of Expertise in an Ill-Structured Domain 772
Colleen M. Zeitz, University of Pittsburgh

Group V: Cognition in Context

V.1. Educational Methods

Learning from Indifferent Agents 780
Lisa Dent, Carnegie Mellon University
Jeffrey C. Schlimmer, Carnegie Mellon University

Using Knowledge Representation to Study Conceptual Change in Students for Teaching Physics 788
C. Franklin Boyle, Carnegie Mellon University
Dewey I. Dykstra, Jr., Boise State University
Ira A. Monarch, Carnegie Mellon University

The Effect of Feedback Control on Learning to Program with the LISP Tutor 796
Albert T. Corbett, Carnegie Mellon University
John R. Anderson, Carnegie Mellon University

Supporting Linguistic Consistency and Idiosyncracy with an Adaptive Interface Design 807
Jill Fain Lehman, Carnegie Mellon University

V.2. Agents, Goals and Constraints
The Cognitive Space of Air Traffic Control: A Parametric Dynamic Topological Model 812
Steven Cushing, Boston University

Models of Neuromodulation and Information Processing Deficits in Schizophrenia 820
David Servan-Schreiber, Carnegie Mellon University and University of Pittsburgh
Jonathan D. Cohen, Carnegie Mellon University and University of Pittsburgh

A Functional Role for Repression in an Autonomous, Resource-Constrained Agent 828
Michael Fehling, Stanford University
Bernard Baars, Wright Institute, Berkeley
Charles Fisher, San Francisco Psychoanalytic Institute

A Goal-Based Model of Interpersonal Relationships 836
Stephen Slade, Yale University

Volition and Advice: Suggesting Strategies for Fixing Problems in Social Situations 844
Eric A. Domeshek, Northwestern University

POSTER PRESENTATIONS

I. Connectionist Models

A Connectionist Approach to High-Level Cognitive Modeling 852
Rainer Goebel, University of Marburg

Fuzzy Implication Formation in Distributed Associative Memory 860
Rick L. Jenison, University of Wisconsin
Gregg C. Oden, University of Wisconsin

Scenes from Exclusive-Or: Back Propagation is Sensitive to Initial Conditions 868
John F. Kolen, Ohio State University
Jordan B. Pollack, Ohio State University

A Computational Model of Attentional Requirements in Sequence Learning 876
Peggy L. Jennings, University of Oregon
Steven W. Keele, University of Oregon

Harmonic Grammar - A formal Multi-Level Connectionist Theory of Linguistic Well-Formedness: An Application 884
Géraldine Legendre, University of Colorado, Boulder
Yoshiro Miyata, University of Colorado, Boulder
Paul Smolensky, University of Colorado, Boulder

PDP Models for Meter Perception 892
Don L. Scarborough, Brooklyn College/City University of New York
Benjamin O. Miller, Brooklyn College/City University of New York
Jacqueline A. Jones, Brooklyn College/City University of New York

II. Machine Simulations and Algorithms

Some Criteria for Evaluating Designs 900
Thomas R. Hinrichs, Georgia Institute of Technology

Ad-Hoc, Fail-Safe Plan Learning 908
Roland Zito-Wolf, Brandeis University
Richard Alterman, Brandeis University

Networks Modeling the Involvement of the Frontal Lobes in Learning and Performance of Flexible Movement Sequences 915
Raju S. Bapi, University of Texas, Arlington
Daniel S. Levine, University of Texas, Arlington

Discovering Grouping Structure in Music 923
Jacqueline A. Jones, Brooklyn College/City University of New York
Benjamin O. Miller, Brooklyn College/City University of New York
Don L. Scarborough, Brooklyn College/City University of New York

III. Knowledge and Problem-Solving

Explanations of Nutritional Concepts: Role of Cultural and Biomedical Theories 931
Malathi Sivaramakrishnan, McGill University
Vimla Patel, McGill University

Ill-Structured Problem Solving in Instructional Design 939
James G. Greeno, Stanford University
Margaret Korpi, Stanford University
Douglas Jackson, Stanford University
Vera Michalchik, Stanford University

Psychological Test of an Algorithm for Recognizing Subjectivity in Narrative Text 947
Gail A. Bruder, State University of New York at Buffalo
Janyce M. Wiebe, University of Toronto

Cross-Domain Transfer of Planning Strategies: Alternative Approaches 954
Bruce Krulwich, Northwestern University
Gregg Collins, Northwestern University
Lawrence Birnbaum, Northwestern University

The Effect of Alternative Representations of Relational Structure on Analogical Access 962
Catherine A. Clement, Eastern Kentucky University

What Should I Do Now? Using Goal Sequitur Knowledge to Choose the Next Problem Solving Step 970
Michael Redmond, Georgia Institute of Technology

A Case-Based Approach to Creativity in Problem Solving 978
Janet L. Kolodner, Georgia Institute of Technology
Theresa Louise Penberthy, Georgia Institute of Technology

Invited Symposia

Action Systems: Planning and Execution 986
Emilio Bizzi (Massachusetts Institute of Technology), *Chair*
Michael I. Jordan (Massachusetts Institute of Technology)

Cognitive Aspects of Linguistic Theory 988
Howard Lasnik (University of Connecticut), *Chair*
David Pesetsky (Massachusetts Institute of Technology), *Chair*
James T. Higginbotham (Massachusetts Institute of Technology)
John McCarthy (University of Massachusetts)

Computational Models of Category Learning 989
Dorrit Billman (Georgia Institute of Technology)
Douglas Fisher (Vanderbilt University)
Mark Gluck (Stanford University)
Pat Langley (NASA Ames Research Center), *Chair*
Michael Pazzani (University of California, Irvine), *Chair*

Designing an Integrated Architecture: The Prodigy View 997
Jaime G. Carbonell (Carnegie Mellon University), *Chair*
Yolanda Gil (Carnegie Mellon University)
Robert Joseph (Carnegie Mellon University)
Craig A. Knoblock (Carnegie Mellon University)
Steve Minton (NASA Ames Research Center)
Manuela M. Veloso (Carnegie Mellon University)

Execution-Time Response: Applying Plans in a Dynamic World 1005
Kristian J. Hammond (University of Chicago), *Chair*
Phil Agre (University of Chicago)
Richard Alterman (Brandeis University)
Reid Simmons (Carnegie Mellon University)
R. James Firby (NASA Jet Propulsion Lab)

Neonate Cognition 1013
Richard Held (Massachusetts Institute of Technology), *Chair*
Jane Gwiazda (Massachusetts Institute of Technology)
Renee Baillargeon (University of Illinois)
Adele Diamond (University of Pennsylvania)
Jacques Mehler (CNRS, Paris, France)

Perception, Computation and Categorization **1017**
Whitman Richards (Massachusetts Institute of Technology), *Chair*
Aaron Bobick (SRI International)
Ken Nakayama (Harvard University)
Allan Jepson (University of Toronto)

Principle-Based Parsing **1020**
Robert C. Berwick (Massachusetts Institute of Technology), *Chair*
Steven P. Abney (Bell Communications Research)
Bonnie J. Dorr (Massachusetts Institute of Technology)
Sandiway Fong (Massachusetts Institute of Technology)
Mark Johnson (Brown University)
Edward P. Stabler, Jr. (University of California, Los Angeles)

Recent Results in Formal Learning Theory **1026**
Kevin T. Kelly (Carnegie Mellon University)
Clark Glymour (Carnegie Mellon University), *Chair*

Self-Organizing Cognitive and Neural Systems **1027**
Stephen Grossberg (Boston University), *Chair*
Ennio Mingolla (Boston University)
Michael Rudd (Boston University)
Daniel Bullock (Boston University)
Gail A. Carpenter (Boston University)

SOAR as a Unified Theory of Cognition: Spring 1990 **1035**
Richard L. Lewis (Carnegie Mellon University)
Scott B. Huffman (University of Michigan)
Bonnie E. John (Carnegie Mellon University)
John E. Laird (University of Michigan)
Jill Fain Lehman (Carnegie Mellon University)
Allen Newell (Carnegie Mellon University), *Chair*
Paul S. Rosenbloom (University of Southern California)
Tony Simon (Carnegie Mellon University)
Shirley G. Tessler (Carnegie Mellon University)

What is Cognitive Neuroscience? **1043**
David Caplan (Harvard Medical School), *Chair*
Stephen M. Kosslyn (Harvard University), *Chair*
Michael S. Gazzaniga (Dartmouth Medical School)
Michael I. Posner (University of Oregon)
Larry Squire (University of California, San Diego)

What's New in Language Acquisition ? **1044**
Steven Pinker (Massachusetts Institute of Technology), *Chair*
Kenneth Wexler (Massachusetts Institute of Technology), *Chair*
Stephen Crain (University of Connecticut)
Myrna Gopnik (McGill University)
Alan Prince (Brandeis University)
Michelle Hollander (Massachusetts Institute of Technology)
John Kim (Massachusetts Institute of Technology)
Gary Marcus (Massachusetts Institute of Technology)
Sandeep Prasada (Massachusetts Institute of Technology)
Michael Ullman (Massachusetts Institute of Technology)

Submitted Symposia

Attracting Attention **1045**
Patrick Cavanagh (Harvard University)
Ken Nakayama (Harvard University)
Jeremy M. Wolfe (Massachusetts Institute of Technology)
Steven Yantis (Johns Hopkins University)
Ann Treisman (University of California, Berkeley), *Chair*

Conceptual Coherence in Text and Discourse **1052**
Arthur C. Grasser (Memphis State University), *Chair*
Richard Alterman (Brandeis University)
Kathleen Dahlgren (Intelligent Text Processing, Inc.)
Bruce K. Britton (University of Georgia)
Paul van den Broek (University of Minnesota)
Charles R. Fletcher (University of Minnesota)
Roger J. Kreuz (Memphis State University)
Richard M. Roberts (Memphis State University)
Tom Trabasso
Nancy Stein

A New Look at Decision Making **1060**
Susan Chipman (Office of Naval Research), *Chair*
Gary Klein (Klein Associates)
John A. Swets (Bolt Beranek & Newman Laboratories)
Paul Thagard (Princeton University)
Marvin S. Cohen (Decision Science Consortium, Inc.)
Judith Orasanu (Army Research Institute and Princeton University), *Chair*

The Study of Expertise: Prospects and Limits 1067
Anders Ericsson (University of Colorado, Boulder), *Chair*
Neil Charness (University of Waterloo)
Yuichiro Anzai (Keio University)
Vimla L. Patel (McGill University), *Chair*
Guy J. Groen (McGill University)
Fran Allard (University of Waterloo)
Jan Starkes (McMaster University)
Keith Holyoak (University of California, Los Angeles)

Author Listing

Beth Adelson
Department of Computer Science
Tufts University
Medford, MA 02155

Phil Agre
Department of Computer Science
University of Chicago
1100 E. 58th Street
Chicago, IL 60637

David W. Aha
Department of Computer Science
University of California at Irvine
Irvine, CA 92717

Subutai Ahmad
International Computer Science Institute
1947 Center Street
Berkeley, CA 94704

Woo-Kyoung Ahn
Department of Psychology
University of Michigan
Ann Arbor, MI 48104

Venkat Ajjanagadde
4260 Chestnut Street, Apt. B8
Philadelphia, PA 19104

Richard Alterman
Computer Science Department
Brandeis University
Waltham, MA 02254

John R. Anderson
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213-3890

Michael A. Arbib
Center for Neural Engineering
University of Southern California
Los Angeles, CA 90089-2520

Yigal Arens
Information Sciences Institute
4676 Admiralty Way, Suite 1001
Marina Del Rey, CA 90292

Judith Avrahami
The Goldie Rotman Center for Cognitive Science in Education
School of Education
The Hebrew University of Jerusalem
Jerusalem 91905
ISRAEL

Bernard Baars
The Wright Institute
2728 Durand Avenue
Berkeley, CA 94704

Norman I. Badler
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104-6389

William Ball
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

Raju S. Bapi
Department of Mathematics
University of Texas
Arlington, TX 76019-9408

John A. Bateman
Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292-6695

Elizabeth A. Bates
Department of Cognitive Science, D-015
University of California at San Diego
La Jolla, CA 92093

Malcolm I. Bauer
Cognitive Science Laboratory
Princeton University
221 Nassau Street
Princeton, NJ 08542

Irving Biederman
Department of Psychology
University of Minnesota
Minneapolis, MN 55455

Lawrence Birnbaum
Department of Electrical Engineering and Computer Science
Northwestern University
Evanston, IL 60201

C. Franklin Boyle
CDEC
Carnegie Mellon University
Pittsburgh, PA 15213

Nick Braisby
Centre for Cognitive Science
University of Edinburgh
2 Buccleuch Place
Edinburgh EH8 9LW
SCOTLAND

Matthew Brand
Institute for the Learning Sciences
Suite 300
1890 Maple Avenue
Evanston, IL 60201

Michael R. Brent
Artificial Intelligence Laboratory
Massachusetts Institute of Technology
545 Technology Square
Cambridge, MA 02139

Brent C.J. Britton
Media Lab, E15-412
Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139

Gail A. Bruder
Department of Psychology
State University of New York at Buffalo
Buffalo, NY 14260

Bruce D. Burns
Department of Psychology
University of California, Los Angeles
Los Angeles, CA 90024

Claire Cardie
Department of Computer and Information Science
University of Massachusetts
Amherst, MA 01003

Roger Chaffin
Trenton State College
Trenton, NJ 08625

David J. Chalmers
Center for Research on Concepts and Cognition
Indiana University
510 North Fess Street
Bloomington, IN 47408

B. Chandrasekaran
Laboratory for Artificial Intelligence Research, Department of Computer and Information Science
The Ohio State University
Columbus, OH 43210

Yam San Chee
Department of Information Systems and Computer Science
National University of Singapore
Lower Kent Ridge Road
Singapore 0511
REPUBLIC OF SINGAPORE

Axel Cleeremans
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

Catherine A. Clement
Psychology Department
Eastern Kentucky University
Richmond, KY 40475-3108

Jonathan D. Cohen
Western Psychiatric Institute and Clinic
University of Pittsburgh
Pittsburgh, PA 15260

Gregg Collins
Northwestern University
Institute for the Learning Sciences
Suite 300
1890 Maple Avenue
Evanston, IL 60201

Albert T. Corbett
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213-3890

Rosalind A. Crawley
Department of Psychology
University of Durham
Durham DH1 3LE
ENGLAND

Steven Cushing
Department of Computer Science B-4
Boston University
755 Commonwealth Avenue
Boston, MA 02215

Judy DeLoache
Department of Psychology
University of Illinois
603 East Daniel Street
Champaign, IL 61820

Lisa Dent
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Eric A. Domeshek
Institute for the Learning Sciences
Northwestern University
Evanston, IL 60201

Barry B. Druhan
Department of Psychology
236 Audubon Hall
Louisiana State University
Baton Rouge, LA 70803

Kevin Dunbar
Department of Psychology
McGill University
1205 Dr. Penfield Avenue
Montréal, PQ H3A 1B1
CANADA

Dewey I. Dykstra, Jr.
Department of Physics
Boise State University
Boise, ID 83727

Gillette Elvgren III
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Jill Fain Lehman
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Brian Falkenhainer
Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304

Anne L. Fay
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213-3890

Michael Fehling
Intelligent Systems Laboratory
321 Terman Center
Stanford University
Stanford, CA 94305-4025

Jerome A. Feldman
International Computer Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704

Christiane Fellbaum
Cognitive Science Laboratory
Princeton University
221 Nassau Street
Princeton, NJ 08542

Charles Fisher
The San Francisco Psychoanalytic Institute
2420 Sutter Street
San Francisco, CA 94115

Kenneth D. Forbus
Qualitative Reasoning Group
Beckman Institute
University of Illinois
405 North Mathews Avenue
Urbana, IL 61801

Dean Foster
Center for Decision Research
Graduate School of Business
University of Chicago
1101 East 58th Street
Chicago, IL 60637

Bradley Franks
Centre for Cognitive Science
University of Edinburgh
2 Buccleuch Place
Edinburgh EH8 9LW
SCOTLAND

George W. Furnas
Bell Communications Research
435 South Street, MRE 2M-397
Morristown, NJ 07960

Dedre Gentner
Psychology Department
University of Illinois at Urbana
6032 E. Daniel Street
Champaign, IL 61820

Edward Gibson
Department of Philosophy
Carnegie Mellon University
Pittsburgh, PA 15213-3890

Vittorio Girotto
Istituto di Psicologia
Centro Nazionale delle Ricerche
Viale K. Marx 15
I-00137 Roma
ITALY

Rainer Goebel
Fachbereich Psychologie
Universität Marburg
Gutenbergstrasse 18
D-3550 Marburg
WEST GERMANY

Robert L. Goldstone
Department of Psychology
University of Michigan
Ann Arbor, MI 48109

Renison J. Gonsalves
Department of Educational Services
Brooklyn College
Bedford Avenue and Avenue H
Brooklyn, NY 11210

Kenneth C. Gray
Department of Psychology
University of Michigan
Ann Arbor, MI 48104

James G. Greeno
School of Education
SU/Education Bldg. 1353096
Stanford University
Stanford, CA 94305

Mary Hare
Center for Research in Language, C-008
University of California at San Diego
La Jolla, CA 92093

Catherine L. Harris
Department of Cognitive Science, D-015
University of California at San Diego
La Jolla, CA 92093

Barbara Hemforth
Fakultät für Psychologie
Ruhr-Universität Bochum
Postfach 102148
D-4630 Bochum 1
WEST GERMANY

James Hendler
Computer Science Department
University of Maryland
College Park, MD 20742

Thomas R. Hinrichs
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332

Douglas R. Hofstadter
Center for Research on Concepts and
Cognition
Indiana University
510 North Fess Street
Bloomington, IN 47405

Susan Hollbach Weber
International Computer Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704

Keith J. Holyoak
Department of Psychology
Pranz Hall
University of California
Los Angeles, CA 90026

Eduard Hovy
Information Sciences Institute
4676 Admiralty Way, Suite 1001
Marina Del Rey, CA 90292

John E. Hummel
Department of Psychology
University of Minnesota
Minneapolis, MN 55455

Lawrence Hunter
National Library of Medicine
Bldg. 38A, MS-54
Bethesda, MD 20894

Douglas Jackson
School of Education
SU/Education Bldg. 1353096
Stanford University
Stanford, CA 94305

Rick L. Jenison
Department of Psychology
University of Wisconsin
Madison, WI 53706

Peggy L. Jennings
Department of Psychology
Oregon Hall
University of Oregon
Eugene, OR 97403

Hollyn M. Johnson
Department of Psychology
University of Michigan
Ann Arbor, MI 48104

Eric K. Jones
Institute for the Learning Sciences
Northwestern University
1890 Maple Avenue
Evanston, IL 60201

Gregory V. Jones
Department of Psychology
University of Warwick
Coventry CV4 7AL
ENGLAND

Jacqueline A. Jones
Department of Computer Science
Brooklyn College
Brooklyn, NY 11210

Jugal K. Kalita
Department of Computer and Information
Science
University of Pennsylvania
Philadelphia, PA 19104-6389

Nancy G. Kanwisher
Department of Psychology
University of California at Berkeley
Berkeley, CA 94720

Yaakov Kareev
The Goldie Rotman Center for Cognitive
Science in Education
School of Education
The Hebrew University of Jerusalem
Jerusalem 91905
ISRAEL

Steven W. Keele
Department of Psychology
Oregon Hall
University of Oregon
Eugene, OR 97403

Angela Kennedy Hickman
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Hyun-Kyung Kim
Qualitative Reasoning Group
Beckman Institute
University of Illinois
405 North Mathews Avenue
Urbana, IL 61801

David Klahr
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213-3890

David Kleinman
Department of Psychology
University of Durham
Durham DH1 3LE
ENGLAND

Boicho Nikolov Kokinov
Institute of Mathematics
Bulgarian Academy of Sciences
Bl. 8, Acad. G. Bonchev Street
Sofia 1113
BULGARIA

John F. Kolen
Laboratory for AI Research, Department of Computer and Information Science
The Ohio State University
Columbus, OH 43210

Janet L. Kolodner
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332-0280

Margaret Korpi
School of Education
SU/Education Bldg. 1353096
Stanford University
Stanford, CA 94305

Chris A. Kortge
Department of Psychology
Stanford University
Stanford, CA 94305

Bernadette Kowalski
Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Bruce Krulwich
Northwestern University
Institute for the Learning Sciences
Suite 300
1890 Maple Avenue
Evanston, IL 60201

John Laird
Artificial Intelligence Laboratory
University of Michigan
Ann Arbor, MI 48105

Jill H. Larkin
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

George Lakoff
International Computer Science Institute
1947 Center Street
Berkeley, CA 94704

David B. Leake
Center for Research on Concepts and Cognition
Indiana University
510 North Fess Avenue
Bloomington, IN 47408

Paolo Legrenzi
Dipartimento di Psicologia
Via dell'Università 7
I-34127 Trieste
ITALY

Wendy Lehnert
Department of Computer and Information Science
University of Massachusetts
Amherst, MA 01003

Géraldine Legendre
Department of Linguistics
University of Colorado
Boulder, CO 80309-0430

Daniel Levine
Department of Mathematics
University of Texas
Arlington, TX 76019-9408

Steven L. Lytinen
Artificial Intelligence Laboratory
University of Michigan
Ann Arbor, MI 48109

Siobhan B.G. MacAndrew
Department of Psychology
University of Warwick
Coventry CV4 7AL
ENGLAND

Arthur B. Markman
Department of Psychology
Beckman Institute
University of Illinois
405 North Mathews Avenue
Urbana, IL 61801

Joel D. Martin
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332

Thomas Mastaglio
Department of Computer Science
University of Colorado
Campus Box 430
Boulder, CO 80309-0430

Robert C. Mathews
Department of Psychology
236 Audubon Hall
Louisiana State University
Baton Rouge, LA 70803

Robert McCartney
Department of Computer Science and
Engineering, U-155
University of Connecticut
Storrs, CT 06269-3155

James L. McClelland
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

Vera Michalchik
School of Education
SU/Education Bldg. 1353096
Stanford University
Stanford, CA 94305

Risto Miikkulainen
Artificial Intelligence Laboratory
Computer Science Department
University of California
Los Angeles, CA 90024

Benjamin O. Miller
Department of Computer Science
Brooklyn College
Brooklyn, NY 11210

Donald H. Mitchell
Amoco Production Company
Tulsa Research Center, P.O. Box 3385
Tulsa, OK 74102

Melanie Mitchell
Center for Research on Concepts and
Cognition
Indiana University
510 North Fess Street
Bloomington, IN 47405

Yoshiro Miyata
Department of Computer Science
University of Colorado
Boulder, CO 80309-0430

Ira A. Monarch
Department of Philosophy
Carnegie Mellon University
Pittsburgh, PA 15213

Stephen T. Morgan
Department of Psychology
University of Wisconsin
Madison, WI 53706

Michael C. Mozer
Department of Computer Science
University of Colorado
Boulder, CO 80309-0430

Kenneth S. Murray
Department of Computer Science
University of Texas at Austin
Austin, TX 78712

N. H. Narayanan
Laboratory for Artificial Intelligence
Research, Department of Computer and
Information Science
Ohio State University
Columbus, OH 43210

Nancy J. Nersessian
Princeton University
Program in History of Science
Princeton, NJ 08544

Sheldon Nicholl
Department of Computer Science
University of Illinois
405 North Mathews Avenue
Urbana, IL 61801

Dan Oblinger
Qualitative Reasoning Group
Beckman Institute
University of Illinois
405 North Mathews Avenue
Urbana, IL 61801

Gregg C. Oden
Department of Psychology
University of Wisconsin
Madison, WI 53706

Stellan Ohlsson
Learning Research and Development Center
University of Pittsburgh
LRDC Building, 3939 O'Hara Street
Pittsburgh, PA 15260

Stephen Omohundro
International Computer Science Institute
1947 Center Street
Berkeley, CA 94704

Christopher Owens
Computer Science Department
University of Chicago
1100 East 58th Street
Chicago, IL 60637

Vimla Patel
Cognitive Studies in Medicine
McGill University
3655 Drummond Street, Rm. 529
Montréal, PQ H3G 1Y6
CANADA

Michael J. Pazzani
Department of Information and Computer Science
University of California
Irvine, CA 92717

Theresa Louise Penberthy
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332-0280

Sandra L. Peters
Department of Computer Science
230 Computing Center
State University of New York at Buffalo
Amherst, NY 14260

Jordan B. Pollack
Laboratory for Artificial Intelligence Research, Department of Computer and Information Science
Ohio State University
2036 Neil Avenue
Columbus, OH 43210

Ian Pratt
Department of Computer Science
University of Manchester
Manchester M13 9PL
ENGLAND

Yulin Qin
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

Alex Quilici
Artificial Intelligence Laboratory
Computer Science Department
3531 Boelter Hall
University of California at Los Angeles
Los Angeles, CA 90024

Clark N. Quinn
Learning Research and Development Center
University of Pittsburgh
LRDC Building, 3939 O'Hara Street
Pittsburgh, PA 15260

Ashwin Ram
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332-0280

William J. Rapaport
State University of New York
Department of Computer Science
226 Bell Hall
Buffalo, NY 14260

Mary Jo Rattermann
Department of Psychology
University of Illinois
603 East Daniel Street
Champaign, IL 61820

Michael Redmond
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332-0280

Brent Reeves
Department of Computer Science
University of Colorado
Campus Box 430
Boulder, CO 80309-0430

Brian J. Reiser
Cognitive Science Laboratory
Princeton University
221 Nassau Street
Princeton, NJ 08542

Ellen Riloff
Department of Computer and Information Science
University of Massachusetts
Amherst, MA 01003

Scott P. Robertson
Department of Psychology
Rutgers University
Busch Campus
New Brunswick, NJ 08903

Lewis G. Roussel
Department of Psychology
236 Audubon Hall
Louisiana State University
Baton Rouge, LA 70803

Peter A. Sandon
Department of Mathematics and Computer Science, Computer Science Program
Bradley Hall
Dartmouth College
Hanover, NH 03755

Don L. Scarborough
Department of Psychology
Brooklyn College
Brooklyn, NY 11210

Jeffrey C. Schlimmer
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Walter Schneider
Learning Research and Development Center
University of Pittsburgh
LRDC Building, 3939 O'Hara Street
Pittsburgh, PA 15260

Lael J. Schooler
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

Christian D. Schunn
Department of Psychology
McGill University
1205 Dr. Penfield Avenue
Montréal, PQ H3A 1B1
CANADA

Erwin M. Segal
Center for Cognitive Science
Department of Psychology
State University of New York
Buffalo, NY 14260

Colleen M. Seifert
Department of Psychology
University of Michigan
Ann Arbor, MI 48104

David Servan-Schreiber
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Emile Servan-Schreiber
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213-3890

Judith M. Shedden
Learning Research and Development Center
University of Pittsburgh
LRDC Building, 3939 O'Hara Street
Pittsburgh, PA 15260

Jeff Shrager
Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304

Thomas R. Shultz
Department of Psychology
Stewart Biological Sciences Building
McGill University
1205 Dr. Penfield Avenue
Montréal, PQ H3A 1B1
CANADA

Glenn Silverstein
Department of Information and Computer Science
University of California
Irvine, CA 92717

Herbert A. Simon
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

Malathi Sivaramakrishnan
Cognitive Studies in Medicine
McGill University
3655 Drummond Street, Rm. 529
Montréal, PQ H3G 1Y6
CANADA

David B. Skalak
Department of Computer and Information Science
University of Massachusetts
Amherst, MA 01003

Stephen Slade
Computer Science Department
Yale University
P.O. Box 2158, Yale Station
New Haven, CT 06520-2158

Steven L. Small
Department of Neurology
322 Scaiffe Hall
University of Pittsburgh
Pittsburgh, PA 15260

Paul Smolensky
Department of Computer Science
Institute of Cognitive Science
University of Colorado
Boulder, CO 80309-0430

James C. Spohrer
Advanced Technology Group
Apple Computer, MS: 76-3A
20450 Stevens Creek Boulevard
Cupertino, CA 95014

Rosemary J. Stevenson
Department of Psychology
University of Durham
Durham DH1 3LE
ENGLAND

Suzanne Stevenson
Department of Computer Science
University of Maryland
College Park, MD 20742

Andreas Stolcke
Department of Electrical Engineering and Computer Science
University of California
Berkeley, CA 94720

Gerhard Strube
Fakultät für Psychologie
Ruhr-Universität Bochum
Postfach 102148
D-4630 Bochum 1
WEST GERMANY

Katia P. Sycara
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

James W. Tanaka
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

John Theios
Department of Psychology
University of Wisconsin
Madison, WI 53706

David S. Touretzky
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Kurt VanLehn
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

Michael R. Waldmann
Institut für Psychologie
Universität Frankfurt
Georg-Voigt Strasse 8
D-6000 Frankfurt-am-Main 11
WEST GERMANY

DeLiang Wang
Center for Neural Engineering
University of Southern California
Los Angeles, CA 90089-2520

Leo Wanner
Project Komet
GMD-IPSI
Dolivostr. 15
D-6100 Darmstadt
WEST GERMANY

Nigel Ward
Computer Science Division
571 Evans Hall
University of California at Berkeley
Berkeley, CA 94720

Kim Weber
Department of Psychology
Rutgers University
Busch Campus
New Brunswick, NJ 08903

Deirdre W. Wheeler
Department of Linguistics
University of Pittsburgh
Pittsburgh, PA 15260

Janyce M. Wiebe
Department of Computer Science
University of Toronto
Toronto, Ontario M55 1A4
CANADA

Mark Wiesmeyer
Artificial Intelligence Laboratory
University of Michigan
Ann Arbor, MI 48105

David C. Wilkins
Department of Computer Science
University of Illinois
405 North Mathews Avenue
Urbana, IL 61801

Garry Wilson
Department of Psychology
University of Durham
Durham DH1 3LE
ENGLAND

Heike Wrobel
Fakultät für Psychologie
Ruhr-Universität Bochum
Postfach 102148
D-4630 Bochum 1
WEST GERMANY

Horng Jyh P. Wu
Artificial Intelligence Laboratory
University of Michigan
Ann Arbor, MI 48105

Berrin A. Yanikoglu
Department of Mathematics and Computer
Science, Computer Science Program
Bradley Hall
Dartmouth College
Hanover, NH 03755

Ilan Yaniv
Center for Decision Research
Graduate School of Business
University of Chicago
1101 East 58th Street
Chicago, IL 60637

Colleen M. Zeitz
Learning Research and Development Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15260

Roland Zito-Wolf
Computer Science Department
Brandeis University
Waltham, MA 02254

Ingrid Zukerman
Department of Computer Science
Monash University
Clayton VIC 3168
AUSTRALIA

EFFECT OF STRUCTURE OF ANALOGY AND DEPTH OF ENCODING ON LEARNING COMPUTER PROGRAMMING

Yam San CHEE
Department of Information Systems and Computer Science
National University of Singapore

Abstract

This research addresses the need for effective ways of teaching computer programming. It focuses on two aspects of instruction. First, the research investigates the use of analogy in teaching programming. It extends existing research by investigating what constitutes a *good* analogy. Second, the research investigates the effect of depth of encoding on programming performance.

The factors *analogy* and *encoding* were manipulated in a 3 x 2 factorial design. Analogy was operationalized by varying the clarity and systematicity/abstractness of the analogies used. Encoding was operationalized by varying the frequency with which deep encoding and elaboration of learned material were invoked by the presentation of questions on the learned material. The dependent variables were score obtained on program comprehension and program composition tasks and the time taken to perform the tasks. Research subjects were 15- to 17-year-olds without prior exposure to computer programming. Differences in mathematics ability and age were controlled.

The results provide empirical support for a predictive theory of the relative goodness of competing analogies. They provide only marginal support for depth of encoding (as operationalized) in learning computer programming effectively. *Post hoc* data analysis suggests that good analogies assist the learning of semantics but not syntax. Furthermore, the effect of encoding was only apparent in learning syntax but not semantics.

Introduction

The traditional approach to teaching computer programming by emphasizing programming language statements (Mayer, 1979; Spohrer & Soloway, 1986) has proved unsatisfactory. Such an approach fails to assist in the acquisition of a useful mental model of the notional machine underlying the programming language (Bayman & Mayer, 1983) and to facilitate the transition from programming knowledge to programming behavior (Anderson, Farrell & Sauers, 1984).

Explanatory Analogy

Analogies are a useful tool for learning and instruction (see, for example, Norman, 1980; Rumelhart & Norman, 1981). The validity of this claim has been demonstrated in the domain of learning computer programming (Mayer, 1975, 1976). Analogies can provide the required mental model of the notional machine. They can also facilitate the transition from programming knowledge to behavior as novices attempt to execute their mental model. Simons (1984) posits that analogies assist learning by making abstract information imaginable and concrete, by providing an existing schema as the basis for the formation of a new schema, and by making relevant anchoring ideas available so that new information can be actively integrated with prior knowledge.

Gentner (1982) postulates the characteristics of analogy that contribute to explanatory power. Her postulation is based on a well-defined theory of structure-mapping (Gentner, 1983) that distinguishes between *attributes* and *relations* on one hand and between *first-order relations* and *higher-order relations* on the other.

An explanatory analogy may be viewed in terms of three properties of internal structure: clarity, richness, and systematicity/abstractness (Gentner, 1982). *Clarity* refers to how base nodes are mapped onto target nodes. A violation occurs if one base node maps to two or more distinct target nodes *or* if two or more distinct base nodes map to the same target node. *Richness* refers to predicate density: that is, for a given set of nodes, the average number of predicates per node that can be plausibly mapped from base to target. *Systematicity/abstractness* refers to the degree to which the imported predicates belong to a mutually constraining conceptual system. Higher-order relations that link lower-order relations are the essence of systematicity. Highly systematic mappings are generally also abstract because they contain a greater proportion of higher-order relations.

The Theory of the Structure of Explanatory Analogies is derived, in part, from distinctions drawn by Gentner (1982) between good and bad explanatory analogies. It states that important, regularly occurring structural differences exist between good explanatory analogies and weak explanatory analogies. In particular, (1) good explanatory analogies possess clarity; weak explanatory analogies do not; (2) good explanatory analogies are higher in systematicity and abstractness than weak explanatory analogies; and (3) good explanatory analogies are lower or equal in richness to weak explanatory analogies.

The theory is used as the basis for distinguishing between the explanatory power of alternative analogies in this research. For achievement in both program comprehension and program composition, learning with an analogy that possesses the structural properties of good explanatory analogy is expected to result in a better

learning outcome than learning with an analogy that possesses the structural properties of weak explanatory analogy.

Depth of Encoding

Learning outcomes depend not only on the quality of instruction but also on the efficacy of cognitive processing *during* the learning phase. Good learning outcomes are associated with depth of encoding (Craik & Lockhart, 1972). Greater depth implies a greater degree of semantic or cognitive analysis on the part of the student, resulting in superior understanding, recall, and retention of material learned. The initial encoding of learned material can pass through further elaboration whereby more associations are formed between newly acquired knowledge and prior knowledge. The establishment of these associations leads to better integration of new knowledge with old knowledge and improved understanding of learned material.

Depth of encoding also results in better recall because of a more persistent memory trace (Craik & Lockhart, 1972), with deeper levels of encoding associated with more elaborate, stronger, and more lasting traces. In addition, retention is a function of depth of encoding, as well as other factors such as the amount of attention devoted to a stimulus and the time available for processing the stimulus.

Based on the foregoing, the quality of students' learning when acquiring knowledge related to a new and unfamiliar domain should be significantly affected by the depth of encoding they achieve during learning. Deeper encoding should be facilitated by presenting instructional material in relatively short segments followed by questions on the material just learned. The presentation of questions forces students to try to actively understand the instructional material so that they can answer the questions correctly. Consequently, deep encoding and elaboration receive active support. The presentation of questions in short segments also eases the burden of learning because a lighter cognitive load is placed upon memory.

Where students complete their study of the entire instruction set *before* attempting questions on the materials learned, depth of encoding is less well supported. The absence of questions that evoke deeper processing of instructional material during learning results in more superficial processing and, consequently, in poorer understanding, poorer retention, and poorer recall ability. Furthermore, when students are required to answer questions only at the end of the instruction set, the cognitive load on memory is very great because students have to draw their answers from across the *entire* instruction set.

Thus, deep encoding is expected to result in better understanding, retention, and recall of learned material, and hence in superior programming task performance compared to shallow encoding that occurs when the entire instruction set is studied before questions on the instruction set are attempted.

Hypotheses Tested

The research hypotheses are based on four theoretical constructs: (1) explanatory power of analogy, (2) depth of encoding, (3) quality of program comprehension, and (4) quality of program composition.

Hypothesis 1 The quality of program comprehension when learning with a good analogy is better than the quality of program comprehension when learning with a weak analogy or without an analogy.

Hypothesis 2 The quality of program comprehension when learning with a weak analogy is better than or equal to the quality of program comprehension when learning without an analogy.

Hypothesis 3 The quality of program comprehension when learning with deep encoding is better than the quality of program comprehension when learning with shallow encoding.

Hypothesis 4 The differences in quality of program comprehension when learning with a good analogy, a weak analogy, and without an analogy will be greater when learning with shallow encoding than when learning with deep encoding; that is, there will be an interaction effect between the explanatory power of analogy and the depth of encoding.

Hypothesis 5 The quality of program composition when learning with a good analogy is better than the quality of program composition when learning with a weak analogy or without an analogy.

Hypothesis 6 The quality of program composition when learning with a weak analogy is better than or equal to the quality of program composition when learning without an analogy.

In general, the above hypotheses follow from the preceding discussion. In Hypothesis 4, an analogy is postulated to possess an *integrating* function in addition to the functions of concretizing, structurizing, and active assimilation. Hypotheses 5 and 6 are similar to Hypotheses 1 and 2 and are based on the expectation that mastery of syntax and semantics is an essential component of program coding ability.

Method

Design The factors *analogy* and *encoding* were manipulated in a 3 x 2 factorial design. Analogy comprised three levels: (1) good analogy, (2) weak analogy, and (3) no analogy (a control condition). Encoding comprised two levels: (1) deep, and (2) shallow. The experiment was conducted in two phases: a program comprehension phase followed by a program composition phase. Two dependent variables were used in each phase. The dependent variables in the

program comprehension phase were (a) program comprehension score, and (b) time taken to answer comprehension questions. The dependent variables in the program composition phase were (a) program composition score, and (b) time taken to answer composition questions. Both the program comprehension and program composition scores are performance metrics obtained by applying a predetermined scoring template to subjects' responses. The experimental design incorporated two covariates: mathematics ability and age.

Subjects Subjects were school students between the ages of 15 and 17 years. They were unexposed to computer programming. Ninety valid subjects' responses were obtained; 60 were boys and 30 were girls. They were assigned randomly to treatment conditions.

Materials

Treatment Materials. The treatment materials comprised three sets of instruction on programming in BASIC: (1) the good analogy set, (2) the weak analogy set, and (3) the no analogy set. In the good analogy set, the instructional materials were woven around an analogy that dealt with a master processor and his three assistants – the assigner, the reader, and the printer – working together in a room to perform the operations of a notional computer. Data were input either through an input slot or via data cards that came through an input window on an input wall. Data were output via an output window on an output wall. Window boxes in the room stored the values of variables whose names were written on the boxes. In the weak analogy set, the underlying analogy was similar but less elaborate. There was only one window through which both input and output were handled. In addition, the names of the three assistants were generalized to "assistant," "messenger," and "helper" in order to facilitate the one-to many and many-to-one object mappings in the weak analogy. Finally, the no analogy set presented the instructional material without reference to any analogy.

The instructional materials covered the program statements LET, PRINT, END, REM, INPUT, DATA, READ, GO TO, and IF/THEN. Looping constructs were taught using the IF/THEN and GO TO statements.

The exact length of the instructional materials was controlled. To compensate for the additional text required to present the analogy material, filler text (which presented a brief history of computers) was added to the weak analogy and no analogy materials so that the word count for each set of instructional materials was identical.

The good and weak analogy treatment materials instantiated the Theory of the Structure of Explanatory Analogies. The instructional materials contained the base of the analogy (good or weak) woven into the instruction on BASIC.

A sample of the good and weak analogies, depicted in propositional network form, is shown at the end of this paper. Networks 1 and 9 depict those portions of the base of the good analogy and weak analogy respectively that deal with the organization of the computer. The corresponding targets of the good and weak analogies are depicted in Networks 5 and 13.

Object mappings between base and target may be inferred directly by the positions the object nodes occupy in two-dimensional space. In the weak analogy networks, however, this method does not apply if an object participates in a one-to-many or many-to-one mapping. In such instances, the object mapping is specifically shown using a striped arrow.

Relation mappings from base to target are inferred via the identical positions that the relations occupy in the two-dimensional space of the propositional networks. Exceptions to this rule again occur in the weak analogy networks, and they occur when object nodes do not map one-to-one from base to target. Unlike object nodes, however, relations that map across always do so with the same name. First-order relations are depicted by normal arrows; higher-order relations are depicted by heavy arrows. Higher-order relations constrain lower-order relations in accordance with structure-mapping theory.

The operationalization of the Theory of the Structure of Explanatory Analogies can be summarized as follows. First, the good analogy possesses clarity because all object mappings from base to target are one-to-one; the weak analogy does not possess clarity because it contains two one-to-three and three one-to-two mappings. Second, the richness (predicate density) of the good analogy is 2.00, and the richness of the weak analogy is 1.85. From a practical viewpoint, richness may be regarded as equal; the closeness of the richness measures is not surprising given that *both* the good and weak analogies aim to explain the operations of the notional computer. Third, the good analogy possesses higher systematicity/abstractness than the weak analogy because 1 third-order relation, 9 second-order relations, and 78 first-order relations were mapped to the target in the good analogy compared with no third-order relations, 6 second-order relations, and 57 first-order relations being mapped across in the weak analogy.

Test Materials. The test materials comprised two sets of questions. The first set was designed to test program comprehension. It was divided into eight parts: (1) Elements of the BASIC language; (2) The replacement statement LET; (3) The PRINT statement; (4) Review: LET, PRINT, and END; (5) The INPUT statement; (6) The DATA and READ statements; (7) The unconditional transfer statement GO TO; and (8) The decision statement IF/THEN.

The second set of questions was designed to test program composition. The set comprised seven questions in increasing order of difficulty and covered the full range of BASIC statements presented.

Procedure The study was conducted in two sessions. Session 1 (the program comprehension phase) commenced at 10:00 a.m. On average, the session lasted 2 1/2 hours. Session 2 (the program composition phase) commenced after a lunch break of about one hour. The session lasted 1 1/2 hours on average.

The encoding factor (deep versus shallow) was operationalized by administering the instructional materials and test questions differently in Session 1. In the deep encoding condition, subjects alternated between reading instructional material on BASIC and answering questions on the material just read. In the shallow encoding condition, subjects read the entire set of instructional materials. They then answered each set of questions in the same order as subjects in the deep encoding condition.

After subjects had been instructed on how the experiment would be conducted, the experiment proper commenced. Subjects were told to begin reading the instructional materials placed before them. As they completed the reading, each raised their hand to indicate to the researcher that they had done so.

If subjects were in the deep encoding condition, they were given the first set of printed questions to answer. The researcher asked them to start work and started the stopwatch. On completion of the set of questions, subjects stopped the stopwatch and raised their hand. The researcher then recorded the time taken on the question sheet. The question sheet was then put away. The next set of instructional materials was then given to the subject. Subjects continued by alternating between reading instructional material and answering questions until the last set of questions was answered.

If subjects were in the shallow encoding condition, they first read the entire set of instructional materials. They then answered each set of questions in the same order and following the same procedure as subjects in the deep encoding condition.

Upon completing Session 1, subjects were released for lunch. When they returned for Session 2, they were instructed on the conduct of the experiment in the program composition phase. They were then given 10 minutes to review the instructional materials they had read in Session 1. The researcher then started each subject on the program composition task and, at the same time, started the stopwatch.

When subjects completed the program composition questions, they stopped the stopwatch and raised their hand. The researcher recorded the time taken on the question sheet. Subjects who completed the experimental task were each paid $20.

Scoring Subjects' responses to the program comprehension and program composition questions were scored according to a template designed by the researcher. The scoring scheme was devised to reward the demonstration of correct knowledge of BASIC and to maximally discriminate between the levels of achievement attained by subjects. An independent check of scoring reliability was performed.

Results and Discussion

Program Comprehension

Figure 1 shows the means for program comprehension score. The program comprehension data were analyzed using ANCOVA and MANCOVA. Hypotheses

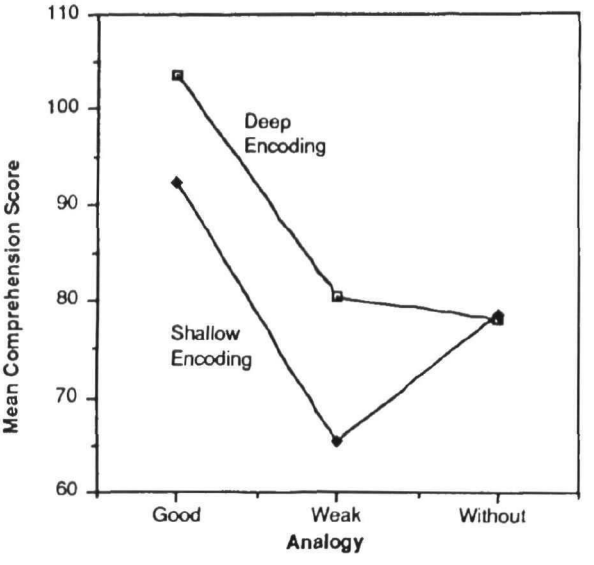


Figure 1

1 and 2 were evaluated using planned comparisons. Hypothesis 1 was confirmed when quality of program comprehension was evaluated in terms of program comprehension score ($p = .001$). It was also confirmed when quality of program comprehension was evaluated in terms of program comprehension score and time ($p = .003$).

Similarly, Hypothesis 2 was confirmed when quality of program comprehension was evaluated in terms of program comprehension score ($p = .499$). The proposition was also supported from a multivariate viewpoint ($p = .395$).

Consistent support for Hypotheses 1 and 2 when evaluated in terms of program comprehension score as well as program comprehension score and time confirms that the quality of program comprehension when learning with a good analogy is significantly better than that associated with learning with a weak analogy or without an analogy, and the quality of program comprehension when learning with a weak analogy and when learning without an analogy are not significantly different. Given the mix of clarity, richness, and systematicity/abstractness operationalized in the experiment, the theory's prediction that these characteristics effectively define good analogy is supported.

Hypothesis 3 was marginally supported when evaluated in terms of program comprehension score ($p = .055$). Despite the lack of statistical significance, the overall (weighted average) program comprehension scores were in the predicted direction (Deep Encoding, $M = 86.5$; Shallow Encoding, $M = 77.4$). The marginal significance of the univariate result may be due the task complexity and constrained experimental learning time that did not allow the expected benefit of deep encoding on the creation and restructuring of knowledge to materialize fully. A deep, semantic appreciation of the notional computer's operations requires that the knowledge acquired be assimilated and restructured over the course of learning. However, restructuring is associated with knowledge understanding but requires time to take effect (Norman, 1978).

Hypothesis 4, the posited interaction between the analogy and encoding factors, was not supported when the quality of program comprehension was evaluated in terms of program comprehension score and also in terms of program comprehension score and time. Thus, the expectation that the analogy would help subjects in the shallow encoding condition more than subjects in the deep encoding condition was not confirmed. Instead, the data suggest that the analogy and encoding factors are independent.

Program Composition

Figure 2 shows the means for program composition score. The program composition data were analyzed using covariance analysis. However, a test of the homogeneity of regression assumption revealed that the assumption was violated for the dependent variable program composition time.

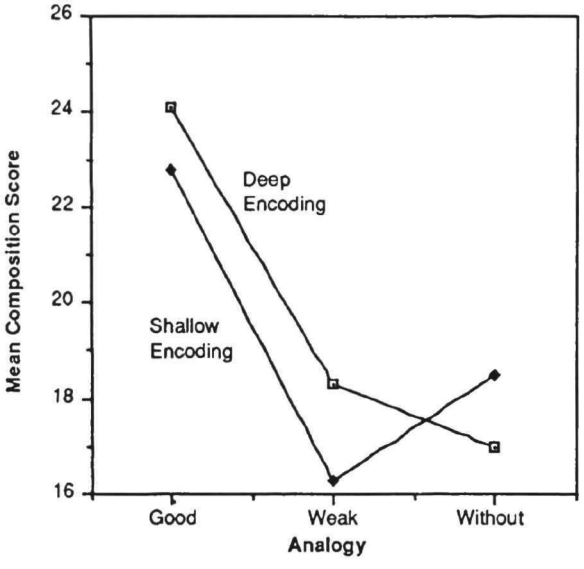


Figure 2

Consequently, the program composition data were analyzed using a covariance model proposed by Searle (1979). *Covariate-adjusted observations* were obtained by rearrangement of the model equation, and the model was estimated using MANOVA.

Using planned comparisons, Hypothesis 5 was supported when evaluated in terms of covariate-adjusted program composition score ($p = .014$). It was also supported when evaluated in terms of covariate-adjusted program composition score and time ($p = .044$). Hypothesis 6 was supported when evaluated in terms of covariate-adjusted program composition score ($p = .998$) and also when evaluated in terms of covariate-adjusted program composition score and time ($p = .613$). The nonsignificant result indicates equality between the weak analogy and no analogy treatment conditions.

The consistent program composition results show that the Theory of the Structure of Explanatory Analogies is also supported with respect to program composition. Note that the analogy factor accounted for 6.6% of the explained variance of program composition score but accounted for 12.0% of the explained variance of program comprehension score. A *transfer of learning* from program comprehension to program composition is thus evident. The smaller effect of type of analogy on program composition is consistent with the expectation that the explanatory power of analogy facilitates performance in program composition via achievement in program comprehension.

Post Hoc Analysis

While scoring subjects' responses to the program comprehension questions, it was noticed that the variability of scores on questions that focused on the syntactic rules of BASIC statements was consistently smaller than it was on questions that focused on the conceptual understanding associated with the operations of the notional computer. This phenomenon suggested that it might be fruitful to investigate the data further by distinguishing between scores on syntax-oriented questions and scores on semantics-oriented questions. Accordingly, the program comprehension data were subclassified into syntax scores and semantics scores and analyzed further via a *post hoc* analysis.

The data for the *post hoc* analysis were analyzed using ANCOVA and MANCOVA. The dependent variables were syntax score and semantics score. The experimental design was identical to that used in the main analysis.

Model estimation revealed that for syntax score, the analogy factor was significant ($p = .015$); the encoding factor was also significant ($p = .012$). For semantics score, however, only the analogy factor was significant ($p = .003$). From a multivariate viewpoint, both the analogy factor and the encoding factor were significant ($p = .001$ and $p = .043$ respectively).

Planned comparisons were performed to evaluate Hypotheses 1 and 2 for syntax, semantics, and both syntax and semantics. For syntax, the good analogy versus weak analogy and no analogy comparison was marginally significant ($p = .056$), while the weak analogy versus no analogy comparison was significant ($p = .027$).

For semantics, the good analogy group was significantly better than the weak analogy and no analogy groups ($p = .001$), and the weak analogy and no analogy groups were not significantly different ($p = .841$). That is, the comparisons were consistent with the results obtained for the composite program comprehension score.

Some interesting insights are obtained from the above analysis. The significance of the analogy factor on both syntax score and semantics score and the significance of the encoding factor only on syntax score suggest that the analogy treatment affects performance on both syntax and semantics, while the encoding treatment affects performance on syntax only.

Furthermore, it becomes clear that the marginal significance of the encoding factor on the composite program comprehension score ($p = .055$) was attributable to the effect of deep encoding on syntax ($p = .012$), not on semantics ($p = .117$). This result suggests that students

learn the technical (rule-like) nature of syntactic knowledge effectively when such knowledge is tested shortly after it is presented. In effect, the quick application of newly-acquired syntactic knowledge assists students in assimilating the rules associated with syntax and helps to drill them in the application of such rules.

By contrast, the lack of significance of the encoding factor on semantics suggests that, contrary to the intended outcome of the encoding treatment, a deep semantic understanding of program statements was not achieved probably because of the limited exposure that students were given to programming. The hypothetical time division associated with complex learning proposed by Norman (1978) suggests that the bulk of knowledge restructuring (and hence deep semantic understanding) occurs during the central phase of learning, after sufficient time has been spent on the accretion of knowledge. Given the restricted learning time in the experiment (approximately four hours), the relatively small amount of time spent on restructuring appears to be due to the use of analogy rather than the use of deep encoding.

Examination of the weighted average means for syntax shows that the good analogy and no analogy groups were more alike than different, while the weak analogy group was unlike both (Good Analogy, M = 25.5; Weak Analogy, M = 20.6; No Analogy, M = 24.1). This result suggests that the weak analogy harmed the acquisition of syntactic knowledge, and the good analogy did not assist the acquisition of such knowledge. However, for semantics, the good analogy group was distinct from the weak analogy and no analogy groups (Good Analogy, M = 72.6; Weak Analogy, M = 52.0; No Analogy, M = 54.6). Thus, the good analogy assisted the acquisition of semantic knowledge but not syntactic knowledge.

Research Conclusions

The Theory of the Structure of Explanatory Analogies was empirically tested. The research supported the theory's prediction that clarity and systematicity/abstractness are structural characteristics of analogy that effectively capture the strength of its explanatory power. *Post hoc* analysis further revealed that good analogy assists the acquisition of semantic programming knowledge but not syntactic programming knowledge.

From the viewpoint of experimental methodology, the *explicit* operationalization and measurement of systematicity and abstractness has shown that these structural characteristics of analogy can be derived objectively and in a manner that possesses empirical validity. Thus, the usefulness of the syntactic perspective on knowledge representation based on the concepts of systematicity and abstractness has been demonstrated.

References

Anderson, J. R., Farrell, R. & Sauers, R. (1984). Learning to Program in LISP. *Cognitive Science*, **8**, 87-129.

Bayman, P. & Mayer, R. E. (1983). A diagnosis of beginning programmers' misconceptions of BASIC programming statements. *Communications of the ACM*, **26**, 677-679.

Craik, F. I. M. & Lockhart, R. S. (1972). Levels of processing: A framework for memory research. *Journal of Verbal Learning and Verbal Behavior*, **11**, 671–684.

Gentner, D. (1982). Are scientific analogies metaphors? In D. S. Miall (Ed.), *Metaphor: Problems and Perspectives*, pp. 106-132. Brighton, Sussex: The Harvester Press.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, **7**, 155-170.

Mayer, R. E. (1975). Different problem-solving competencies established in learning computer programming with and without meaningful models. *Journal of Educational Psychology*, **67**, 725-734.

Mayer, R. E. (1976). Some conditions of meaningful learning for computer programming: Advance organizers and subject control of frame order. *Journal of Educational Psychology*, **68**, 143-150.

Mayer, R. E. (1979). A psychology of learning BASIC. *Communications of the ACM*, **22**, 589-593.

Norman, D. A. (1978). Notes toward a theory of complex learning. In A. M. Lesgold, J. W. Pellegrino, S. Fokkema & R. Glaser (Eds.), *Cognitive Psychology and Instruction*, pp. 39-48. NY: Plenum.

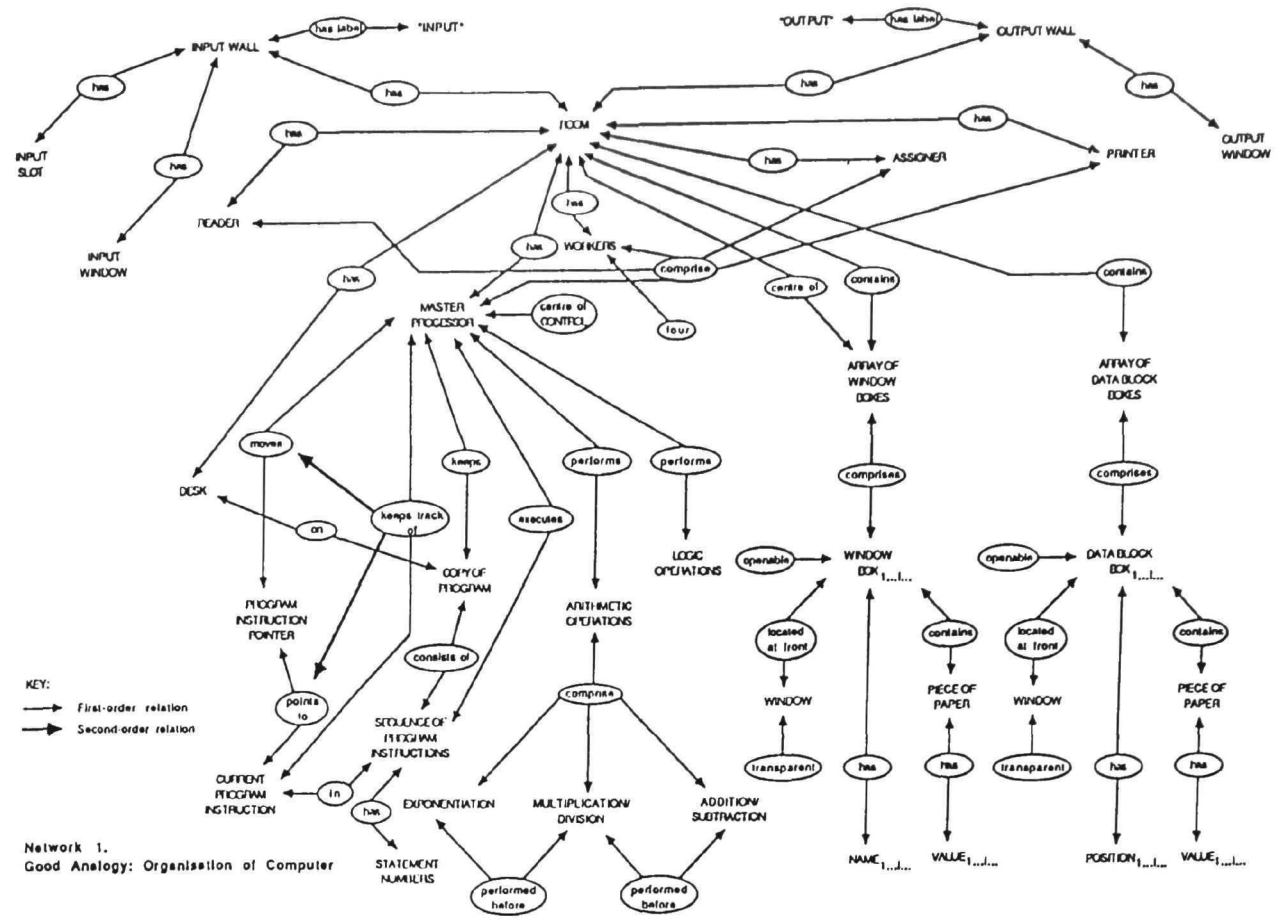
Norman, D. A. (1980). Teaching, learning, and the representation of knowledge. In R. E. Snow, P. A. Frederico & W. E. Montague (Eds.), *Aptitude, Learning, and Instruction*, Volume 2, pp. 237-244. Hillsdale, NJ: Lawrence Erlbaum Associates.

Rumelhart, D. E. & Norman, D. A. (1981). Analogical processes in learning. In J. R. Anderson (Ed.), *Cognitive Skills and their Acquisition*, pp. 335-359. Hillsdale, NJ: Lawrence Erlbaum Associates.

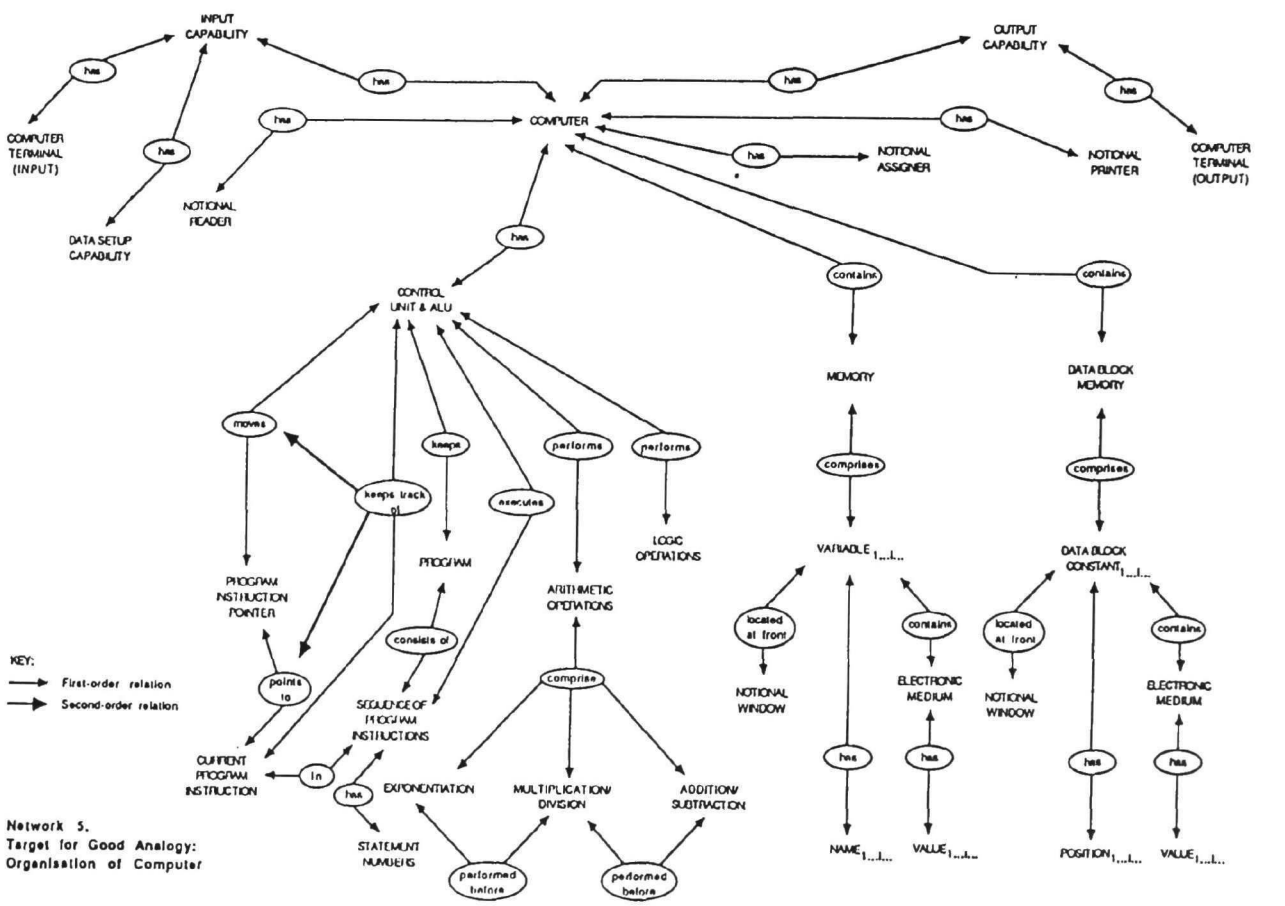
Searle, S. R. (1979). Alternative covariance models for the 2-way crossed classification. *Communications in Statistics – Theory and Methods*, **A8**, 799-818.

Simons, P. R. J. (1984). Instructing with analogies. *Journal of Educational Psychology*, **76**, 513-527.

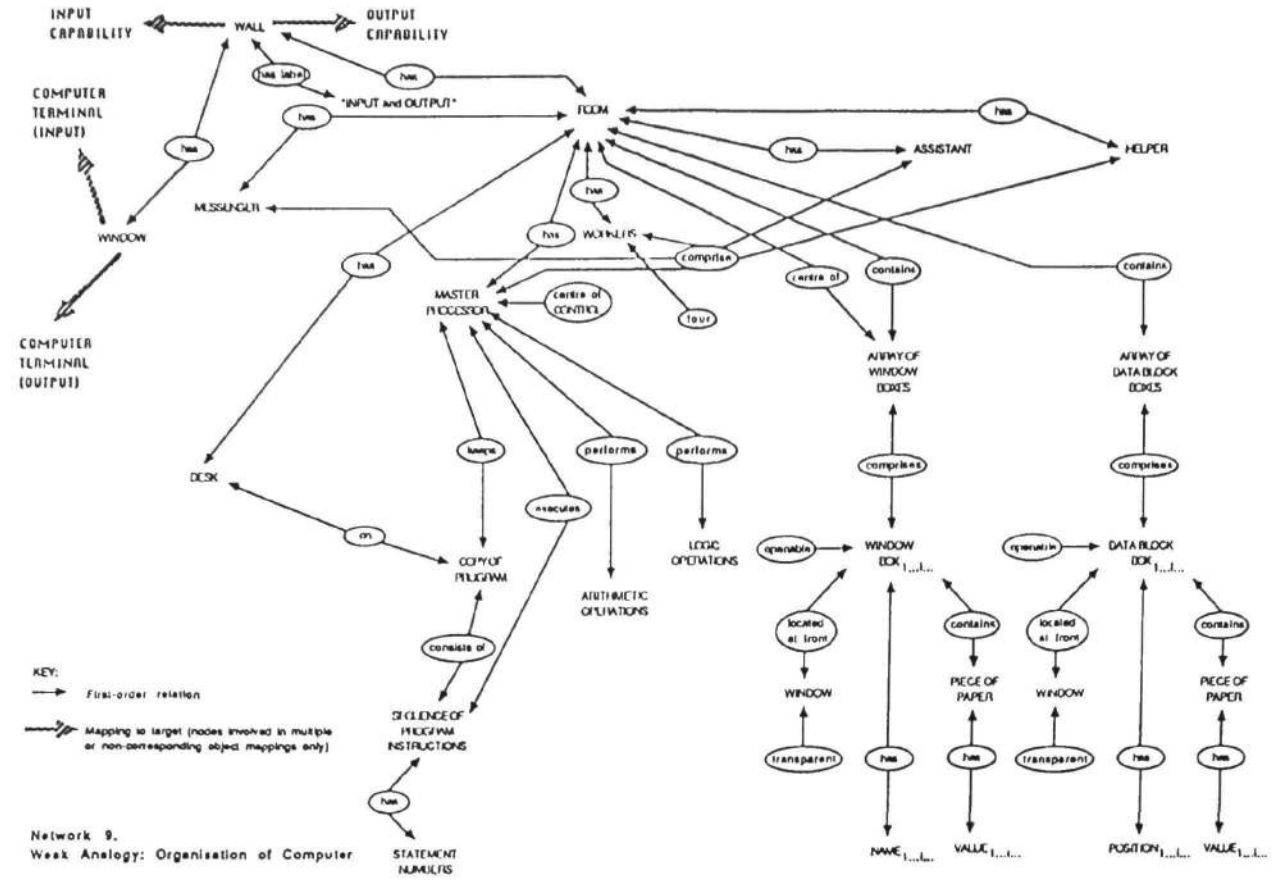
Spohrer, J. C. & Soloway, E. (1986). Novice mistakes: Are the folk wisdoms correct? *Communications of the ACM*, **29**, 624-632.



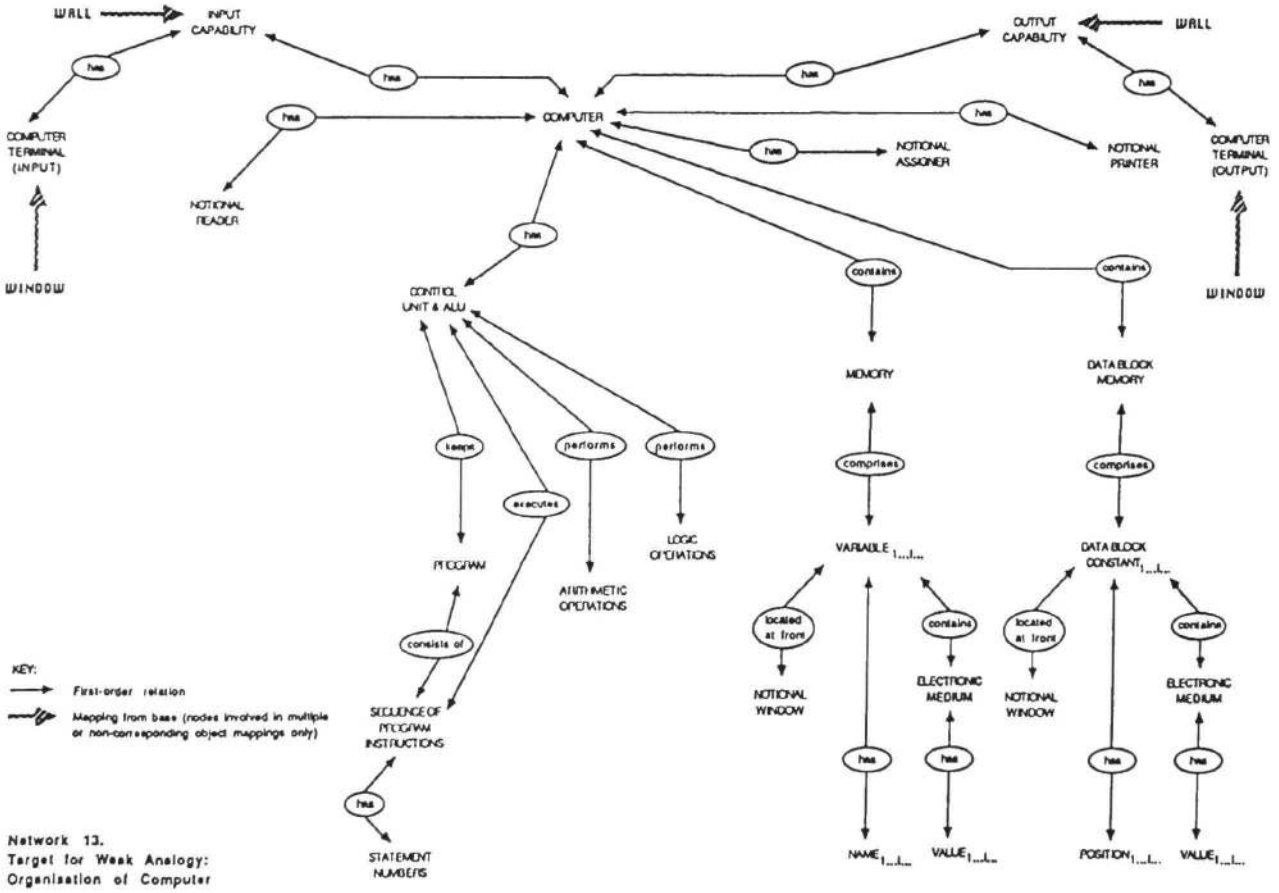
Network 1.
Good Analogy: Organisation of Computer



Network 5.
Target for Good Analogy:
Organisation of Computer



Network 9.
Weak Analogy: Organisation of Computer



Network 13.
Target for Weak Analogy:
Organisation of Computer

Evaluating and Debugging Analogically Acquired Models

Beth Adelson
Dept. of Computer Science
Tufts University
Medford, MA 02155 [1]

Abstract

We describe elements of a cognitive theory of analogical reasoning. The theory was developed using protocol data and has been implemented as a computer model. In order to constrain the theory, it has been developed within a problem-soving context, reflecting the purpose of analogical reasoning. This has allowed us to develop: A purpose-constrained mapping process which makes learning and debugging more tractable; An evaluation process that actively searches for bugs; And a debugging process that maintains functional aspects of base models, while adding target-appropriate causal explanations. The active, knowledge-based elements of our theory are characteristic of mechanisms needed to model complex problem-solving.

1. Introduction: Motivation and Goals

The research described here is part of an attempt to develop a cognitive theory of analogical reasoning. Because the purpose of analogical reasoning is to learn and solve problems, we have developed our theory within a problem-solving context. This approach provides powerful insights into the phenomenon and constraints on the theory.

Recent research suggests a class of theories which rests on the processes of retrieval, mapping, evaluation, debugging and generalization (Carbonell, 1983 & 1986; Falkenhainer, Forbus & Gentner, 1986; Holyoke & Thagard 1985 & in press; Burstein & Adelson 1987; Kolodner, 1985; Kedar-Cabelli, 1984). Our work extends existing cognitive theories of analogical reasoning by specifying the processes of mapping, evaluation and debugging as active; constrained by the problem-solving context; and dependent on knowledge about function, structure and mechanism [2]. In our theory, the mapping process can be focussed so that partial domain models, sufficient for the immediate problem-solving purpose, can be mapped. This incremental learning strategy renders both mapping and debugging more tractable (Adelson & Burstein, 1987). A detailed description of our theory of mapping can be found in Adelson (1989), here we focus on evaluation and debugging:

1. Active Evaluation: An intelligent problem-solver needs to be able to identify the bugs inherent in an analogically acquired domain model. Our system searches for bugs in newly mapped domain models. It does so by comparing the nature of the actions and objects in the newly mapped model to the nature of the actions and objects appropriate in the domain being mapped into. This allows the system to identify the aspects of the model that are inappropriate and therefore unlikely to hold in the domain being mapped into. Our system also has knowledge about the way in which analogical correspondences are meant to be understood across domains. These aspects of our system reflect powerful elements of human reasoning.

2. Active Debugging: Once the buggy portion of a model has been identified it must be replaced by a representation that is accurate in the new domain. Our system constructs and runs simulations in both source and target domains in order to identify mechanisms with *analogous functionality*. This allows the system to correct mapped models, maintaining the functionality provided by the analogical example while building a representation of a mechanism appropriate to the target domain. Here too our system's behavior characterizes the constrained way in which analogical examples are understood and used.

2. Protocol Data Illustrating the Issues

[1] Thanks to Michael Brent and Mike Futeran. Also thanks to Dedre Gentner, Brian Falkenhainer and Ken Forbus for their loan of SME.

This work was funded by Carnegie-Mellon's NSF funded EDRC and by a grant from NSF's Engineering Directorate.

[2] The spirit of Carbonell's (1986) work on derivational analogy; Holyoke & Thagard's (in press) work on multiple constraint satisfaction; Kedar-Cabelli's (1984) work on purpose-guided reasoning, Burstein's (1983) work on causal reasoning and Falkenhainer, Forbus & Gentner's (1986) work on structure mapping is consonant with our view of analogical reasoning as an active, knowledge intensive process.

In developing our theory we have repeatedly drawn on the protocol data described below. These data yield insights into processes that need to be described in implementing a cognitive problem-solver.

The Protocol:

In collecting our data we video-taped a tutor teaching a student about stacks. The tutor's goal was to have the student be able to write Pascal procedures for pushing and popping items on to and off of stacks. At the beginning of the protocol session the student had just completed an introductory programming course in which he had learned about some basic programming constructs and about elementary data structures such as arrays and simple linked lists. (He had not learned about using a linked list as a stack.) The tutor had the intention of building upon the student's existing knowledge of Pascal through the use of analogy.

The relevant events of the protocol can be summarized as follows:

Learning about the behavior of a stack:

The tutor told the student that stacks are so named because their behavior is analogous to the behavior of the device that holds plates in a cafeteria. The student then proceeded to think of ways in which he might have previously encountered the use of stacks in programming; he suggested that stacks might be useful in implementing subroutine calls. He then stated that, in general, when a task had an unmet precondition it would be useful to delay execution of the task by pushing it onto a stack.

Learning about the mechanism underlying the behavior:

The tutor told the student that the mechanism of the computer science stack is in some sense analogous to the mechanism of the cafeteria stack. In order to achieve 'last in first out' (LIFO) behavior items are pushed *and* popped at the top of the cafeteria stack. The student drew a diagram of a cafeteria stack and noted that *push* causes the stack's spring to compress and *pop* causes it to expand.

Implementing push and pop in the target domain:

The student then wrote the code for *push* and *pop*. After writing *push*, however, the student asked if the capacity limitation which results when the spring is fully compressed is relevant in the new domain. The tutor told the student that the *physical* elements of the analogy (springs, movement of plates, etc.) do not apply. The student then asked if the concept of capacity limitation applies even if the spring doesn't. The tutor responded that although capacity limitation is an important concept, the student should disregard it for now.

3. Issues for Specifying a Theory of Analogical Problem-Solving

As diagrammed in Figure 1, the class of model to which our system belongs, contains a mapper, an evaluation and debugging mechanism, and a problem-solving component. The mapper takes as input a base domain model and a list of the correspondences between elements in the base model and already known target elements. The mapper produces tentative target domain models which are then debugged and evaluated. The debugged models are then used to in problem-solving. Additonally, in our system, the output of the debugger can be used to guide subsequent mappings.

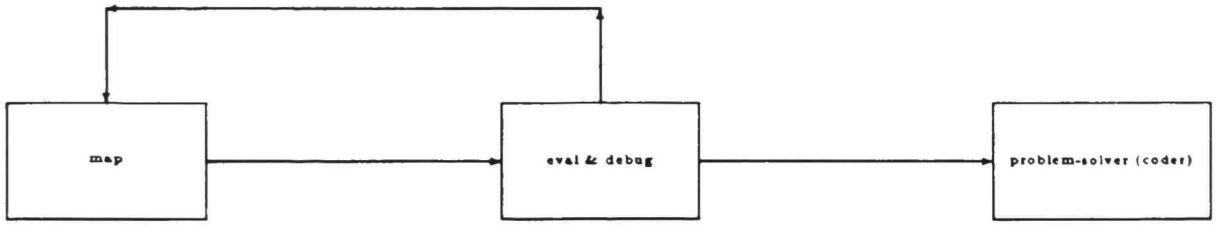


Figure 1: Components of the Analogical Reasoner

3.1. Purpose-Constrained Mapping

Purpose provides an essential constraint in problem-solving, but current implementations of cognitive theories do not make sufficient use of this constraint[3]. The argument for why purpose is necessary in constraining

[3] Thagard and Holyoke (in press), and Kedar-Cabelli, (1986) also stress the theoretical importance of purpose.

human problem-solving runs as follows. Understanding a complex domain requires understanding a number of distinct aspects of the domain and the relationships among those aspects (Burstein, 1986; Collins &Gentner, 1983; Adelson, 1984). Given the constraints of the cognitive system, it is not possible to learn all of these various aspects at one time. Rather, to make learning of a complex domain more tractable, students and instructors typically focus on individual, purpose-related aspects of the domain and, one at a time, map partial models from more familiar analog domains (Burstein & Adelson, 1987 & in press)[4]. In our computational description, the learning process starts with this selection and mapping of purpose-constrained aspects of the target domain. Our mapping mechanism focuses on partial models of the base domain whose type reflects the problem solver's purpose, and maps these models *separately*, type by type, over to the target domain[5].

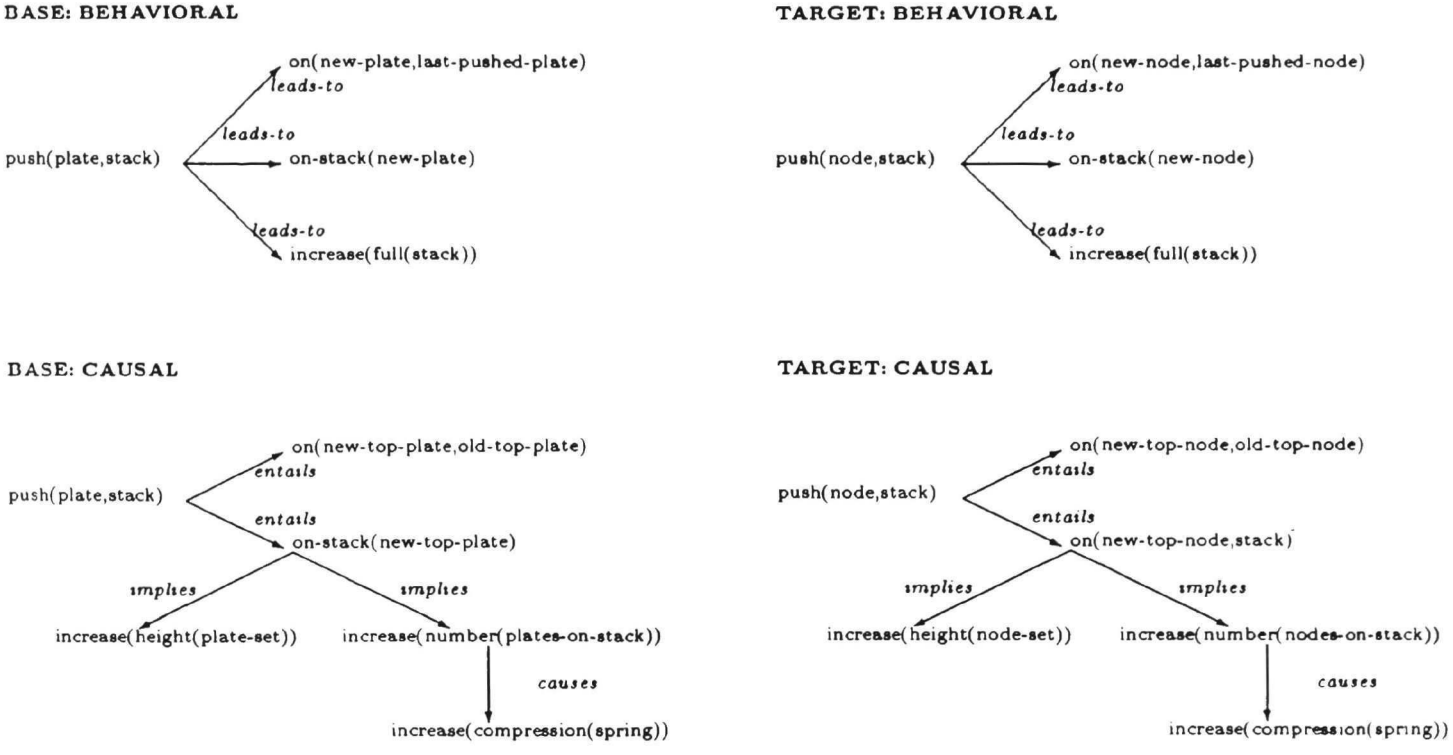


Figure 2: Behavioral and Causal Models (top and bottom) in the Base and Target Domains (left and right)

The following illustrates a mapping produced by our system:
In this example our overall goal is to have the system model the problem-solving in our protocol. That is, we want the system to learn about computer science stacks and stack operations by analogy to cafeteria stacks and to produce the code for the operations *push* and *pop*. To accomplish this, the system begins by following the tutor's suggestion to focus initially on the behavioral model for *push*[6]. It selects that model from a base domain containing behavioral and causal models of both *push* and *pop*. The selected model is then mapped into the target domain. The behavioral and causal models for *push* in the base domain can be seen in the left half of Figure 2 (top and bottom respectively).

In the upper right of the figure we see the behavioral model of the target domain produced by our mapping mechanism. What is important to note here[7] is that the nature of the models allows them to be used in the problem-solving that is the system's ultimate goal. That is, the models in the figure describe the chains of events

[4]Our protocol data illustrates this process. The student acquired a sophisticated understanding of the behavior of a stack before learning about the mechanism supporting the behavior.

[5]See Burstein & Adelson (in press) for an explanation of the difference between models of varying types.

[6]The tutor also supplies a list of base and target domain correspondences stating for example, that *push* in the base corresponds to *push* in the target and plates correspond to nodes.

[7]Space limitations force us to omit several points about the models. The assertions in the model can be formalized to allow, for example, the deduction that the pushed node becomes the new top node. Additionally, the predicates do have an underlying

that occur when domain operations are performed (Schank & Riesbeck, 1981; Schank & Abelson, 1977). As a result, they can be run on the system's *simulation machine* (Adelson, 1989; Forbus, 1985; de Kleer & Brown, 1985). This allows them not only to be examined by a debugger (Sections 3.2.1 and 3.2.2) but also to be used to generate Pascal code after debugging.

The following section describes how our system debugs the newly mapped model of *push* so that problem-solving can be carried out successfully.

3.2. Debugging a Newly Mapped Model

3.2.1 Actively Seeking out Bugs

In our theory, debugging is characterized by an *active* search for bugs. The base domain model is known, by definition, to provide an imperfect model of the target domain. The base model may contain inappropriate elements that require deletion or transformation; or it may require additional knowledge specific to the target domain.

Our current example illustrates the case in which a newly mapped model contains a concept that is inappropriate in the target domain. Looking at the behavioral model in the target domain (Figure 2, upper right) we see that it pushing a node onto a stack which is implemented as a list of nodes, leads to the stack being more full. However, the system contains prior knowledge about the target domain which asserts that lists of nodes are used when a data structure without a pre-specified capacity limitation is desired[8]. Since linked lists have no specified capacity limitation there is an inconsistency between the newly mapped model and prior knowledge of the target domain. The system must have the ability to notice and resolve this inconsistency.

Identifying and Fixing Bugs in a Runnable Model:

Here we describe how the system's evaluation and debugging mechanism resolves the 'fullness' bug in the course of evaluating the behavioral model of *push*. The system's *evaluator* traverses a newly mapped model in an attempt to determine whether each element it encounters is appropriate in light of the domain the model has been mapped into. In order to allow the evaluator to carry out the evaluation the system has been given several kinds of knowledge:

K1. Any element that occurs in a model has a definition, a *template* consisting of a set of features (Burstein, 1983; Waltz, 1982; Winston, 1977 & 1982). Elements in the model are either objects (e.g. *stack*) or predicates which describe attributes of, or actions on the object (e.g. *fullness* or *push* respectively). For objects, one feature in the template specifies the class it belongs to. For predicates the class of both the predicate and its arguments are listed. For example, the predicate *full* is defined as a measurement of the capacity of some argument which must be a limited-capacity container.

K2. The system knows not only which objects and predicates are appropriate to each domain, but also which *classes* of objects and predicates. For example, the system knows that integer variables in particular, and data structures in general, are appropriate in the computer domain.

K3. The system has general knowledge about how analogical correspondences are meant to be taken. For example, the system contains knowledge that physical contiguity in the base can be appropriately thought of as corresponding to virtual contiguity in the target.

The system uses the knowledge described above in applying rules which allow it to evaluate each element in the newly mapped model. The rules are:

R1 Infer that an element currently in the target domain is appropriate in the new model.

R2. If the element is not currently in the target domain but it is of a *class* currently in the target domain, infer that a modified version of the element is appropriate in the new model and use existing domain information

semantics. For example, the value of full results from dividing the current number of plates by the maximum number of plates. Along these lines, the models in the figure have substantially less detail than the actual models used by the system.

[8] For this example we have supplied the system with the same knowledge of the target domain that our novice programmer had. We have given it models for performing typical operations on variables, arrays and linked lists. It also has world knowledge about boxes and containers.

about the class to modify the element[9].

R3. If the element belongs to a class that has a corresponding class in the target (point K3), infer that a modified version of the element belongs and use existing domain information about the coresponding class to modify the model.

R4. A predicate can only be applied to an argument of an appropriate type. That is fullness cannot be predicated of a container without capacity limitations (point K1).

We see each of the above rules (and the knowledge they embody) being applied as we follow the evaluator tracing through the model for *push*. Starting at the root of the tree, the evaluator encounters the element *push*. Because the tutor had specified that *push* in the base corresponded to an asserted, but as yet undefined, version of *push* in the target the system infers that *push* is appropriate to the model and turns to the predicates that follow from it.

The template for the predicate *on* states that it is a "physical contiguity relation" The system knows that physical contiguity in the base corresponds to virtual contiguity in the target (Rule R3). It therefore makes this change to the predicate's template and then infers that the predicate holds. This is an example of the system's ability to interpret analogical correspondences in an appropriate, non-literal manner.

The evaluator comes to the next predicate that *push* leads to. Although *on-stack* does not yet exist in the target, its definition states that it is a "membership relation". The system finds that other predicates in the target are membership relations (e.g. *in-set*) and therefore hypothesizes that the predicate holds (Rule R2).

The evaluator turns to the predicate *full*, it finds that *full* is potentially appropriate in that it already exists in the target as knowledge that arrays can be full (Rule R1). However, the evaluator finds that fullness can only be predicated of containers having capacity limitations (Rule R4). It knows that the stack is being implemented as a list of nodes and that lists do not have capacity limitations. The system suggests that the concept fullness should be removed from the model. It then removes fullness. At this point, if the system is told that the problem arose because no capacity limited containers were being used in this example it will also remove all other predicates whose definitions involve capacity limitations.

But more mileage can be gotten out of this evaluation. The system has just mapped and debugged the behavioral model. It can now to go back and map the causal model using information gained in debugging the behavioral one. When this mapping begins the system will take note of any elements that have been deleted from the behavioral model (in this example, *full*); pieces of the causal model that only support an element already deleted from the behavioral will not be mapped. As a result of our strategy of incrementally mapping partial models and using earlier mappings to guide subsequent ones, the debugging of potentially complex causal models can be made considerably simpler.

3.2.2 Transforming a Mapped Model: Reasoning About Simulations

In the example just described we considered the case where an element needs to be deleted. Our system also handles the case in which a model needs to be corrected by being *transformed* through finding additional correspondences between elements of the base and target domains.

The case in which a model needs this type of transformation is illustrated by the example in which the goal is to implement a stack using an array rather than a linked list. The representation of the base domain is the same as it was for our earlier example. Prior knowledge of the target domain still consists of information about variables and arrays but not about linked lists. The system again has runnable models for typical operations such as initialization and search.

The behavioral model of push is again mapped into the target domain. This time no changes are made in the behavioral model; the fullness of the stack is found to be consistent with the system's knowledge of the capacity limitation of an array. After mapping the behavioral model, the system maps the causal model of the stack into the target domain and then begins to evaluate and debug it. During this process the system questions the tutor on the appropriateness of the spring in the causal model (see Figure 2, bottom right). The tutor tells the system that the domain-appropriate *functional analog* of the spring needs to be found. In finding the functional analog

[9]Space constraints preclude a description of some of the methods that have been developed to modify 'almost right' elements.

of the spring the system will draw on several types of *relational knowledge* which comprise the system's causal ontology[10]:

RK1. The system contains functional to structural mappings; knowledge relating state changes and the mechanisms causing them (Adelson, 1984; de Kleer & Brown, 1985; Forbus, 1985; Kuipers, 1985). For example, it knows that 'changes in fullness are supported by changes in the mechanism comprised of the spring, the set of plates, etc'[11].

RK2. The system has knowledge relating actions and the state changes they produce. It knows, therefore that 'pushing leads to changes in fullness'

RK3. The system also has knowledge relating actions and the mechanisms involved. It knows that 'pushing involves a change in the spring'

In order to find the piece of target domain mechanism with the same function as the spring, the system will find what sort of state change in the base is associated with a particular change in the spring. It will then turn to the target look at the parallel state change and determine what piece of mechanism is effected in the way that the spring was. To do this the system first needs to focus on the base and find what state changes the spring is involved in. It examines its knowledge of functional to structural mappings (RK1) and finds that the spring is involved in changes in fullness. Now, in order to find out the nature of the relationship between changes in fullness and changes in the spring, the system looks for a simulation in which both fullness (RK2) and the spring (RK3) will change. It finds that simulating *push* will produce the needed information. The simulation is run, providing the system with values for the fullness of the stack and the compression of the spring before and after the simulation is run. The system then compares the direction of change in both fullness and spring compression and finds that there is a positive relationship between the two[12].

The system now needs to find what piece of mechanism in the *target* domain changes for the same reason and in the same way as the spring (i.e. increases with fullness). The system begins by looking for an operation in which fullness increases. It will then run this operation and look for pieces of mechanism that register increases in fullness.

Target domain knowledge about the relation between actions and state changes (RK2) asserts that initializing an array causes fullness to increase; the system simulates the process and finds that in the target, it is the array-index that increases with fullness. As a result of this process, in which corresponding simulations are sought, run and evaluated for the purpose of finding functionally analogous mechanisms the system correctly hypothesizes that the array index is the analog of the spring[13]

Space limitations prevent further description of our system, but the debugging process does not end here. Now that the models mapped from the base domain have had changes made to them, they must be checked to see that they are still sufficient. This is done through a series of simulations designed to test that the models still exhibit aspects of LIFO behavior that the system knows are important. For example, pushing and then popping a set of elements must result in reversing their ordering. Additionally, the system must eventually produce both box and arrow and pascal versions of push and pop (Adelson et al, 1988).

4. Summary & Conclusions

we have presented a discussion of three of our system's mechanisms: one for mapping, one for evaluating mapped models and one for debugging inconsistencies. We have implemented a purpose-constrained mapper that reflects the way students limit their focus of attention. The strategy results in incremental learning which makes both mapping and debugging more tractable. We have also implemented an evaluation mechanism that identifies inconsistencies as elements of newly mapped models are checked to see if they are the sort of elements that are known to exist in the target domain. In doing so the evaluation mechanism uses knowledge about the nature

[10] These relations are learned in that the system notices and stores these types of relations whenever it acquires a new model.

[11] The system's knowledge does not contain any explicit statement concerning *how* changes in fullness are related to changes in the spring. This is what needs to be determined.

[12] Currently the system can recognize positive and negative correlations, as well as the lack of relationship between two state variables. It is possible to expand this part of the system to include the recognition of more complex, but regular relationships.

[13] When more than one piece of mechanism is found, the system has the ability to use functional information decide on the better analog.

of the base and target domains and they way in which relations apply across analogous domains. Finally, we have presented a debugging mechanism that maintains functional aspects of base models while adding target-appropriate causal explanations. The development of the mechanisms has been possible because we have worked within a problem-solving context, reflecting the purpose of analogical reasoning.

5 References

Adelson, Beth. Cognitive modeling: Uncovering how designers design. *The Journal of Engineering Design*.. Vol 1,1. 1989.

Adelson, Beth. When novices surpass experts: How the difficulty of a task may increase with expertise *Journal of Experimental Psychology: Learning, Memory and Cognition*, July 1984.

Adelson, B., Gentner, D., Thagard, P., Holyoak, K., Burstein, M., and Hammond, K. The Role of Analogy in a Theory of Problem-Solving *Proceedings of the Eleventh Annual Meeting of the Cognitive Science Society*, 1988.

Burstein, Mark H. Causal Analogical Reasoning. *Machine Learning: Volume I*. Michalski, R.S., Carbonell, J.G. and Mitchell, T M. (Ed.), Los Altos, CA: Morgan Kaufman Publishers, Inc., 1983.

Burstein, Mark H. Concept Formation by Incremental Analogical Reasoning and Debugging. *Machine Learning: Volume II*. Michalski, R.S., Carbonell, J.G. and Mitchell, T M. (Ed.) Morgan Kaufmann Publishers, Inc., Los Altos, CA 1986.

Burstein, M. and Adelson, B. Mapping and Integrating Partial Mental Models. *Proceedings of the Tenth Annual Meeting of the Cognitive Science Society*, 1987.

Burstein, M. and Adelson, B. Analogical Reasoning for Learning. in *Applications of Artificial Intelligence to Educational Testing*. R. Freedle (Ed.) In press. Erlbaum: Hillsdale, NJ.

Carbonell, Jaime G. Transformational Analogy Problem Solving and Expertise Acquisition. *Machine Learning: Volume I*. Michalski, R.S , Carbonell, J.G. and Mitchell, T.M. (Ed.), Los Altos, CA: Morgan Kaufman Publishers, Inc., 1983.

Carbonell, Jaime G. Derivational Analogy: A Theory of Reconstructive Problem Solving and Expertise Acquisition. *Machine Learning: Volume II*. Michalski, R.S., Carbonell, J.G. and Mitchell, T.M. (Ed.), Los Altos, CA: Morgan Kaufman Publishers, Inc., 1986

de Kleer, J. and Brown, J. S. A Qualitative Physics based on Confluences In *Qualitative Reasoning about Physical Systems* D Bobrow editor, MIT Press 1985.

Falkenhainer, B., Forbus, K. and Gentner, D. The Structure-Mapping Engine In *Proceedings of AAAI-86*. Los Altos, CA: Morgan Kaufman, 1986.

Forbus, Ken. Qualitative Process Theory In *Qualitative Reasoning about Physical Systems* D. Bobrow editor, MIT Press 1985

Gentner, Dedre. Structure-Mapping: A theoretical framework for analogy. *Cognitive Science*, 1983, 7(2), 155-70.

Holyoke, K. and Thagard, P. Analogical Mapping by Constraint Satisfaction. Cognitive Psychology. In Press.

Kolodner, J. In *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*. Boulder, CO: Cognitive Science Society, 1985.

Kuipers, B. Commonsense Reasoning About Causality. In *Qualitative Reasoning about Physical Systems* D. Bobrow editor, MIT Press 1985.

Kedar-Cabelli, S. Analogy with Purpose in Legal Reasoning from Precedents. Technical Report 17. Laboratory for Computer Science, Rutgers. 1984.

Schank, R., and Abelson, B. Scripts, Plans, Goals and Understanding. Hillsdale, NJ: Erlbaum, 1977.

Schank, R. and Riesbeck, C. Inside Computer Understanding. Erlbaum: Hillsdale, NJ. 1981.

Waltz, D. Event shape diagrams. In Proceedings of the National Conference on AI. 1982.

Winston, P. Learing new principles from precedents and exercises. *AI*. 1982.

Analogical Process Performance

Clark N. Quinn
Learning Research and Development Center[1]
University of Pittsburgh

Abstract

Analogy is one of the primary mechanisms of cognition, particularly in problem-solving and learning. However, people do not use analogies very effectively. I postulate seven separate processes for analogy that could be responsible for weak analogical reasoning and test those processes independently. The results suggest that performance on analysis of the problem and performance on confirmation of the appropriateness of the analogy both might be suspect in analogical deficits.

Various analyses of the component processes of analogy in problem-solving have been performed. Clement (1981, 1986) has investigated how subjects approached new problems by forming analogies to old ones and has derived process specifications from empirical observation. Sternberg (1977a, 1977b) has specified the steps in analogy for problems where the base domain is known. Holland, Holyoak, Nisbett, and Thagard (1986) have developed a model of induction that includes an account of analogical processes. Gentner (1989) has also outlined a theory of processes in analogy. In contrast to the psychological approaches, Hall (1989) has developed a synthesis with which to analyze the various artificial intelligence models of analogy. These specifications are summarized in Table 1.

Clement	Gentner	HHNT	Hall	Sternberg	Quinn
		Represent		Encode	Analysis
Generate	Access	Select	Recognize		Access
	Map	Map	Elaborate	Map	Map
Confirm	Soundness		Evaluate	Justify	Confirm
					Transfer
Predict				Apply	Solve
Transfer	Store	Transfer	Consolidate		Transfer

Table 1.

Comparison of these analyses of analogy indicates that none of the above specifications includes all the steps that can be involved in the use of analogies. Under Quinn in Table 1 is the analysis proposed for consideration here. This analysis includes an exhaustive list of the processes involved in analogical problem solving. These steps are:

1. *analyze* the target problem to be understood;
2. *access* a familiar base domain;
3. *map* relations between the base and the target;
4. *confirm* the analogy;
5. *transfer* the problem statement of the target to the base domain representation;
6. *solve* the problem for the base domain;
7. *transfer* the solution to the target problem.

---

[1]Based on research performed while the author was at the University of California, San Diego

The first four steps may need to be repeated until a base analogy is successfully accessed and confirmed as being an appropriate model for the target. Some parts of the elaboration of the representations resulting from later processes may actually be accomplished by preceding processes, for instance the transfer of the problem from the target to the base domain may be accomplished as a result of the access and mapping processes. This still requires specification of all the processes, however, as no process can be assumed to be completely subsumed.

This model for analogy leads to predictions about performance. Prior research (Gick & Holyoak, 1980, 1983) has suggested, in a coarse analysis, that accessing an adequate analogy is less likely than successful exploitation of a provided analogy. Which of the cognitive processes of analogy are entailed in obtaining a relevant analogy and which in the subsequent use? Further, which of the processes involved in accessing an analogy might contribute to deficits in performance?

A fine-grained view of access should include the processes of analysis of the problem, access of a base domain, a mapping between the two domains, and confirmation of the analogy. Use of a given analogy, however, should incorporate the processes of mapping between the domains, transfer of the problem to the familiar base domain, solution of the problem in the base domain, and transfer to the target of the solution. Gick & Holyoak found that, given a hint to use a previous problem, subjects are quite adept at using the suggested analogy to generate an acceptable solution. This suggests that the processes involved in use of an analogy, mapping, transfer, and solution, should be adequate.

On the other hand, Gick and Holyoak found that subjects were unlikely to recall a recently presented analogical solution. Which processes might be inadequate in the access of an analogy? When presented with analogies, subjects correctly rate good analogies as better than poor ones (Gentner & Landers, 1985; Ratterman & Gentner, 1987). This would indicate that the subjects can confirm analogies adequately. Mapping has already been identified as a process that should be well practiced and effective. The problem then is to decide whether it is the analysis process or the access process that introduces the performance deficits. The analysis process yields a representation of the target problem that is then used as a basis for the selection of the base domain. The access process is determined by the representation of the target problem. Gentner (1982) has effectively argued that the access is "ballistic" in the sense that the base domain accessed is wholly determined by the representation and once the access process is launched it proceeds without possible intervention to produce a base domain representation. Given a surface representation of the target domain, the access process should return a base domain that matches on surface features. Similarly, if the analysis process results in a deep representation of the problem, the base accessed should return a useful deep analogy. This suggests that analysis is the culpable process. The expectation is that performance on analysis is inadequate. Other processes should have adequate performance.

Experiment: Component Process Performance

This experiment tested the processes independently to determine which processes were culpable in inefficient analogical performance. The processes tested were analysis, access, mapping, confirmation, transfer, and solution.

Method

The processes were tested by having the subject perform the appropriate subsequent process on a problem with the previous processes performed. For example, a subject might be instructed to perform the mapping between one domain, elaborated by the analysis process, and a new domain from the access process.

Each of the six processes was tested in six domains. Testing each process in order or in random order on the same or different domains could lead to contamination effects. For this reason the process factor had to be conducted between subjects. Each subject could perform the process in each of the six domains. This led to a two factor design with one between-subjects factor (task) and one within-subjects factor (domain).

Background knowledge could strongly influence performance on any of the analogical processes. Ideally, subjects would be presented with suitable artificial material so that all subjects share equal knowledge of the base domain and the target. A more practical answer may be to have a variety of analogies over a spectrum of domains, and to allow the subjects to self-evaluate their knowledge of the domain. However, a pilot study (Quinn, 1989) assessed and revealed no effect of self-rated domain knowledge on performance.

Materials. The materials consisted of a workbook containing the six problems. Since there were six different tasks that were administered between subjects, there were six different types of books. Each book consisted of six randomly-ordered problems, each problem in a different domain, all testing the performance of a single process.

I obtained or created six analogies, each with a paired target and base domain. These analogies are drawn from the literature, from my own experience, or were created for the experiment. They cover a broad range of likely experience, from some that require very specific knowledge to some that are likely to be familiar to most every subject.

Within each analogy, I created questions that addressed the specific processes required for analogy. For analysis, subjects were given the target problem and asked *not* to solve the problem, but, rather, merely to analyze the problem, performing all the steps necessary to solve the problem. If the subjects *thought* they had an answer, they were asked to perform the steps that justified their answer. In the access task, the subjects read a target problem and were asked to think of a similar problem. They were instructed that the specifics of the situation might be very similar or widely different. For the mapping task, subjects were given both a target and base domain, with the target domain elaborated, and asked to elaborate the base domain and to establish the correspondences between the two domains. Subjects were instructed to confirm an analogy between two given domains by rating the quality of the match and, more important, justifying their decision. In the transfer task, subjects were given both target and base domains and a particular situation in one domain. They were then asked to find the equivalent situation in the other domain. Transfer was examined as a single process since the cognitive processes operating on the two representations should be perform equivalently in either direction. The transfer process task was balanced between the two directions: transfer of the problem to the base domain and transfer of the solution back to the target domain.The final process tested was solution of a problem. Subjects were given a problem in the base domain and asked to find the solution.

Subjects. The subjects were 86 students in a cognitive psychology course. Participation in the experiment was voluntary. The topic of analogy was part of the course content, but was not presented before the administration of the experiment. These students were predominantly upper-division college students. While their generally high level of college experience *could* conceivably create a pattern of excellent performance on analogical processes, no explicit training is typically encountered in the curriculum and, in fact, their performance on some processes was less than perfect. Subjects with incomplete workbooks were eliminated from analysis. This eliminated 16 of the 86 subjects, leaving a total of 70 subjects for analysis: 11 in analysis, 8 access, 13 mapping, 13 confirmation, 12 transfer, and 13 solution.

Scoring. The data collected from this experiment consisted of six written answers for each subject. Each question was from a different domain. The subjects' answers differ in qualitative ways from the ideal answer. They can range from the subject having performed the wrong task or not having performed any task at all to an essentially perfect performance. Important distinctions between these two extremes are having performed the task but incompletely or poorly or having performed the task adequately but not exceptionally. These four distinctions were assigned a numerical score from one to four: a one (1) represented either no performance or performance of the wrong task, a two (2) was assigned for performing the correct task but not so as to allow the acceptable performance of subsequent processes, a three (3) was assessed for adequate performance of the task allowing subsequent processes to perform correctly, and a four (4) indicated performance of the task including extra performance that indicated an exceptional comprehension of the task.

This conceptual scoring system had to be interpreted differently for each task. While the criteria to determine whether the subject had performed either not at all or the wrong task were

clear, the other performance levels had to have requirements specific for each task. For analysis, to receive an adequate evaluation, subjects had to either re-represent the problem, specify the causal structure, or list the possible solutions. Adequate performance on access required the subjects to access a base domain that had a causal structure that matched the target domain. For acceptable mapping performance, subjects had to determine the corresponding elements in the base domain for all the elements that were elaborated from the target domain. Adequate confirmation was based on using the deep structure of the two domains to evaluate and justify the confirmation decision. Performing transfer to an acceptable level consisted of interpreting the equivalent perturbation in one domain given a modification to the other domain. Finally, a workable solution was required to determine the ability of the solution process. In all domains, performance less than this level resulted in a rating of a two (or a one if a different procedure was performed) while a four was assigned for elaboration on a task beyond the acceptable level.

Results

Coding. Two independent raters scored the data. One rater scored all the responses while a second rater performed a validity check. Several revisions of the rating process led to a procedure producing reliability greater than ninety percent.

Analysis. One of the original six domains was found inadequate because of an incomplete specification of the confirmation task and was eliminated. This left five domains for the six processes.

Process	n	Mean	SD
Analysis	55	2.2364	0.6657
Access	40	2.8500	0.8638
Map	65	3.0462	0.6233
Confirm	65	2.6923	0.7484
Transfer	60	3.4333	0.6475
Solve	65	3.4615	0.6393

Table 2.

A two factor mixed analysis of variance was performed, with six levels of the between subjects factor (process) and five levels of the within subjects factor (domain). There was a significant effect of process $F(5,65)=15.108$, $p<.001$ but no effect of domain $F(4,260)=.81$, $p=.519$.

Performance on analysis was the worst, followed by confirmation and then access. Mapping was adequate and both transfer and solution were performed quite well (see Figure 1).

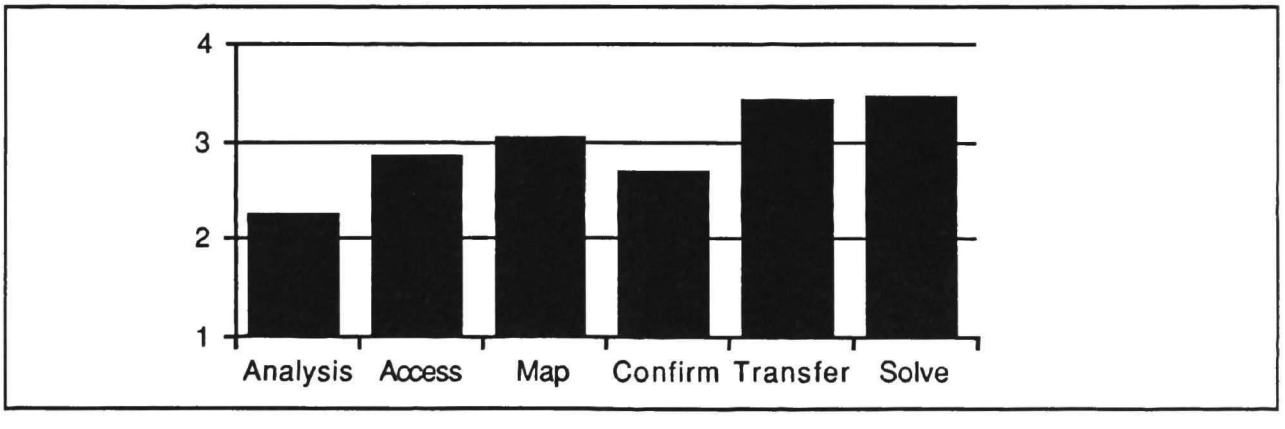


Figure 1.

The interaction between process and domain was significant $F(20,260)=3.779$, $p<.001$. This makes the main effect of process harder to interpret. However, as can be seen from Figure 3, there are a few isolated sources that constitute the majority of the interaction. Two of these sources are the good performance on analysis $t(53)=-2.317$, $p=.012$ and the weak performance on access $t(38)=2.894$, $p=.0031$ for the fifth pair of domains. Overall, patterns indicated by the means are maintained. The interaction can be seen in Figure 2.

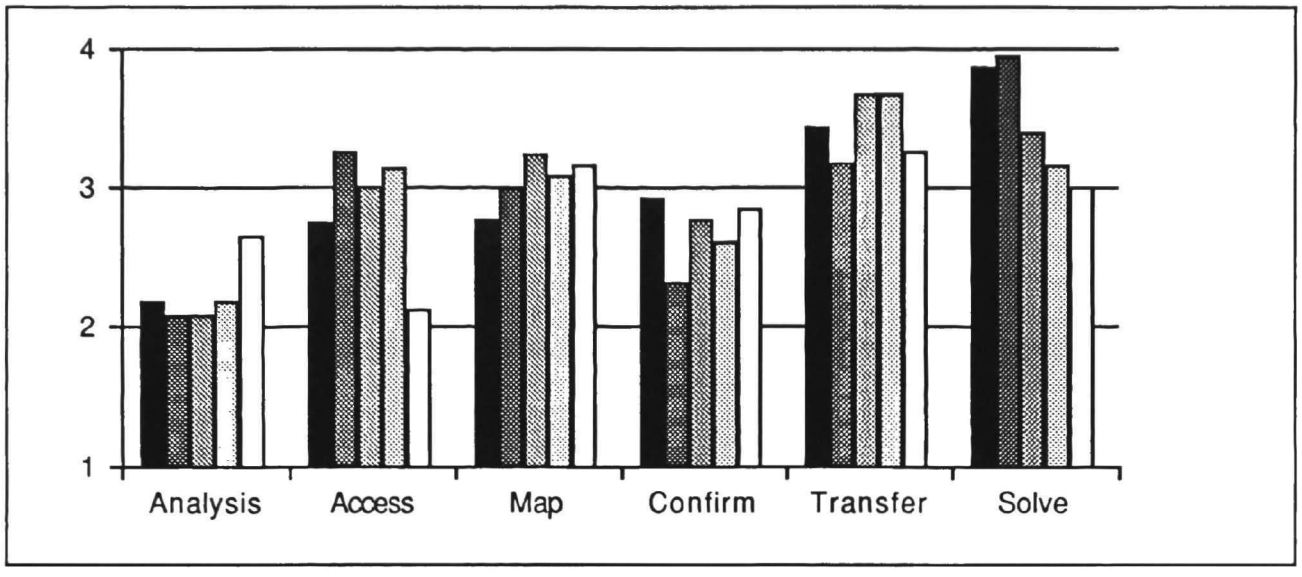


Figure 2.

Discussion

My original hypothesis was that performance on analysis should be inadequate, while performance on other processes should be adequate. These predictions were partially born out by the results of the experiment. Performance on analysis was less than adequate. Unexpectedly, the performances on access and confirmation were also less than adequate. This might be a result of one unusually low domain score for access, but it also might reflect a natural difficulty with access. Perhaps the specification of the analysis in the problems was not enough to ensure adequate access.

Performance on mapping was, on average, of a level sufficient to allow the subsequent processes to apply while performance on transfer and solution were above this level. Performance on solution was surprising, considering that problem-solving behavior has typically been considered weak. The answer may lie in the familiar domains that serve as the base problem for solution.

The result on confirmation is more surprising and harder to explain. Certainly, the task for confirmation here is different than in prior studies. In this study, subjects must explain their confirmation rating, and it is the explanation, not the rating, that receives evaluation. In the previous studies by Gentner (Gentner & Landers, 1985; Ratterman & Gentner, 1987), the subjects were evaluated on their judgement of the quality of the match. Justification may be a more complex and difficult task. Whether or not subjects actually understand the quality of the analogy, they may not be able to express that judgement well.

Another unanswered question about confirmation is whether subjects would use low-rated analogies. The experiments only determined how good the subjects thought the analogies were. One possibility is that subjects may ignore information on the quality of a match in their use of analogies, accepting inadequate base domains whether or not they have the ability to judge them. The cognitive overhead in evaluating the quality of a match, despite any ability to perform the evaluation, prevents this step from being accomplished. Another possibility is that the subjects simply use too low a threshold to confirm analogies. The subjects may have some evaluation of the analogies, and can recognize the relative quality of analogies, but accept the weak analogies.

Again, it might be that the processing load of returning to the analysis and access steps requires too much effort.

Despite the unanswered questions, the performance indications for the identified processes support both the existence of these processes as components of analogy and the utility of the process approach as a framework within which to view analogy. The use of the process approach succeeded in partially succeeded in predicting weak processes of analogy and serves as a guide within which to conduct more discussion about analogical performance deficits.

References

Clement, J. (1981). Analogy generation in scientific problem solving. *Proceedings of the Third Annual Meeting of the Cognitive Science Society*. Berkeley, CA.

Clement, J. (1986). Methods for evaluating the validity of hypothesized analogies. *Proceedings of the Eighth Annual Meeting of the Cognitive Science Society.*

Gentner, D. (1982). *Structure-mapping: a theoretical framework for analogy*. ONR report 5192.

Gentner, D. (1989). Mechanisms of analogical learning. In S. Vosniadou & A. Ortony (Eds.) *Similarity and analogy in reasoning and learning*. London: Cambridge University Press.

Gentner, D., & Landers, R. (1985). Analogical reminding: a good match is hard to find. *Proceedings of the International Conference on Systems, Man and Cybernetics*. Tucson, AZ.

Gick, M. L., & Holyoak, K. J. (1980). Analogical problem solving. *Cognitive Psychology*. 12, 306-355.

Gick, M. L., & Holyoak, K. J. (1983). Schema induction and analogical transfer. *Cognitive Psychology*. 15, 1-38.

Hall, R. P. (1989). Computational approaches to analogical reasoning: a comparative analysis. *Artificial Intelligence*. 39, 39-120.

Holland, J. H., Holyoak, K. J., Nisbett, R. E., & Thagard, P. R. (1986). *Induction: processes of inference, learning and memory*. Cambridge, MA: The MIT Press.

Holyoak, K. J, and Thagard, P. (1989) Analogical mapping by constraint satisfaction. *Cognitive Science*, 13, 3.

Quinn, C. N. (1989). *Analogical process performance and training*. Unpublished doctoral dissertation.

Ratterman, M. J., & Gentner, D. (1987). Analogy and similarity: determinants of accessibility and inferential soundness. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society.*

Sternberg, R. J. (1977a). *Intelligence, information processing, and analogical reasoning*. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Sternberg, R. J. (1977b). Component processes in analogical reasoning. *Psychological Review*. 84, 4, 353-378.

The Effects of Familiar Labels on Young Children's Performance in an Analogical Mapping Task*

Mary Jo Rattermann
University of Illinois
Dept. of Psychology

Dedre Gentner
University of Illinois
Dept. of Psychology

Judy DeLoache
University of Illinois
Dept. of Psychology

Abstract

This research investigates the role of language in children's ability to perform an analogical mapping task. We first describe the results of a simple mapping task in which preschool children performed poorly. In the current study, we taught the children to apply relational labels to the stimuli and their performance improved markedly. It appears that relational language can call attention to domain relations and hence improve children's performance in an analogical mapping task.

A computer simulation of this mapping task was performed using domain representations that differed in their degree of elaboration of the relational structure. The results of the simulation paralleled the experimental results: that is, given deeply elaborated representations, SME's preferred interpretation produced the correct mapping response, while when given shallow representations its preferred interpretation produced an object similarity response. Taken together, the empirical and computational findings suggest that development of analogy and similarity may be explainable in large measure by changes in domain representation, as opposed to maturational changes in processing. They further suggest that relational language may be an important influence on this development.

Introduction

One of the developing child's major achievements is the acquisition of language. This acquisition process pervades almost every aspect of the young child's daily life. Our question in this research concerns the possible effects of language on one aspect of the child's developing abilities: the use of object similarity and relational similarity. Children and adults perform very differently in tasks which require the use of object similarity and/or relational similarity. For example, when given a metaphor such as "A cloud is like a sponge" young children (five years old) produced similarity comparisons based on common object-attributes (e.g., "they both are round and fluffy") while adults produced similarity comparisons based on common relational structures ("they both store water and then later give it back to you") (Gentner, 1988).

This and related developmental differences have led many researchers to suggest that young children use an inherently different mode of processing than adults. Piaget (Piaget, Montangero & Billeter, 1977) proposed that children lack the basic cognitive competence to perform an analogical mapping between objects. This ability is dependent upon cognitive

This work was supported by Center for the Study of Reading contract NIE 400-81-0030. The authors would like to thank Art Markman, Rebecca Campbell, and Jennifer Glenn for their help in the preparation of this paper, and Phyllis Koenig for the wonderful penguins.

structures and processes which do not emerge until they reach the formal operations period of development (approximately 14 years of age). Others propose that children are not "fundamentally different kinds of thinkers" than adults but rather that it is deficiencies in children's knowledge that limits their performance (Brown, 1989; Carey, 1984). The domain-knowledge account would emphasize that metaphor and analogy tasks like the ones described above require knowledge that young children may not possess or may not reliably represent: for instance, knowledge of the causal relations within the two domains.

In this research we sought to (1) trace possible changes in children's ability to perform relational mappings; (2) to investigate whether any such changes could be explained in terms of changes in domain representations and (3) in particular, whether use of relational labels would play a causal role. We wanted to study the child's ability to extract relational similarity from a situation in which other solutions are in principle possible. Therefore, we designed a task in which object similarity was pitted against relational similarity. We then observed whether the child would carry out the relational mapping between the two structures. To further investigate the effects of object similarity, we manipulated the degree of similarity in the object matches by varying the perceptual richness and distinctiveness of the stimulus (Tversky, 1977). With this task we established that preschool age children have difficulty focussing on relational similarity when there is a competing object similarity. We then asked whether language can help children extract relational similarity under these conditions. We first review the basic task and then discuss the language manipulation we used to try to improve children's performance.

The Basic Task

We presented three- and four-year-old children and adults with a simple mapping game in which both object similarity and relational similarity were manipulated (Rattermann, Gentner & DeLoache, 1989). The child and the experimenter each had a set of three objects (clay pots or blue plastic boxes) which increased in size along a continuum from left to right. (See Figure 1.).

Figure 1a.
SIMPLE

Figure 1b.
RICH

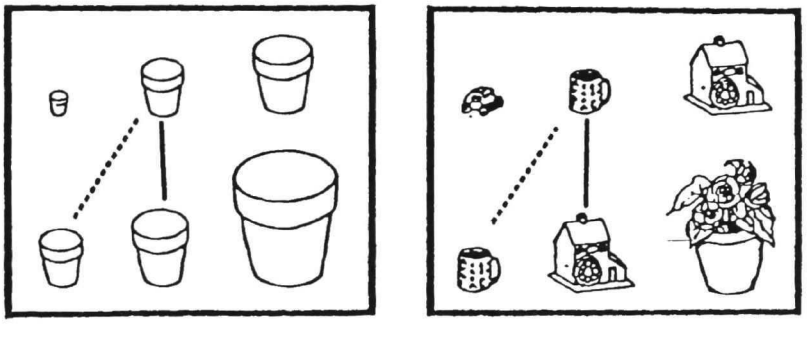


Figure 1. Stimulus sets used in mapping task.
The objects in Row 1 are the experimenter's set and the objects in Row 2 are the child's set.

The experimenter and the child played a game in which the experimenter hid a sticker under the child's set and the child tried to find it. The child was told that if he watched the experimenter as she placed a sticker under one of the objects in her set, he could use the hiding place of her sticker as a clue to the location of his own sticker. We introduced a tension between object similarity and relational similarity by staggering the size of the objects within the triads, creating both a possible object match and a possible relational match (a cross-mapping of the stimuli (Gentner & Toupin, 1986)). That is, if the experimenter's set contained objects of size 1, 2, and 3 the child's set contained objects of size 2, 3, and 4. The task was designed so that the relational response was always correct: the correct response was always based on relative size (e.g., largest object to largest object) and relative position.[1] The child was shown the correct answer and if correct was allowed to keep the sticker. In Figure 1a, the solid line represents the correct relational response, which the child will make if he is able to align the two structures relationally, while the dotted line represents the object-based response which the child will make if he responds on the basis of object similarity rather than relational similarity.

We found an age shift in the performance of this task. The three- and four-year-old children performed quite poorly (an average of 47% relational responses across both ages), while the adults performed extremely well (an average of 87% relational responses). We also found the effect of stimulus richness predicted by Tversky's contrast model; the children performed significantly better with the simple stimulus objects (an average of 54% relational responses for the three-year-olds and 62% for the four-year-olds) than with the rich stimulus objects (an average of 32% relational responses for the three-year-olds and 38% for the four-year-olds), suggesting that the presence of rich, distinctive object matches creates a salient alternative to the relational response (at least for young children). In contrast, when simple objects are used, the resulting object similarity matches are less compelling and therefore less likely to make a competitive alternative to the relational response.[2]

Can Language Promote a Relational Focus?

A growing body of research has investigated the hypothesis that young children use words to focus attention on certain kinds of information. (Gelman & Markman, 1987; Waxman & Gelman, 1986). Gelman and Markman (1987) investigated the role of common word labels on three- and four-year-old children's willingness to extrapolate characteristics between objects. They presented children with a picture of a standard object, e.g., a bluebird, and taught the children a characteristic of this object (e.g. "feeds its baby mashed up food."). The children were then shown a set of several objects, some which shared perceptual similarity

1. The relations of relative size and relative position were perfectly correlated. That is, the middle-sized object was also the object in the middle position.

2. Adults performed roughly equal with the rich and the simple stimuli, suggesting that they can focus on relational commonalities relatively independent of object similarity.

with the standard and some which shared category membership (and therefore a common label) with the standard. When no labels were used the children, as expected, extended the characteristic to objects on the basis of shared perceptual similarity with the standard (e.g, a blue butterfly). When these new objects shared a category label with the standard (e.g. a blackbird) the children extended the characteristic based on the common label and, to a lesser extent, the shared perceptual similarity.

Given this evidence suggesting that labels can direct children's attention to taxonomic object concepts, the question we posed was whether relational labels can direct children's attention to relations. In particular, could the use of relational labels in the perceptual-mapping task influence children to respond relationally. To label the key relative-size relation we chose to use simple, familiar labels: "Daddy", "Mommy," and "Baby". "Daddy", "Mommy," and "Baby" are very salient relations to young children; in fact, children in the previous study occasionally used these labels spontaneously.[3] If the use of relational labels leads children to perform the mapping task correctly, this will support the position that developmental improvement can be accounted for by changes in representation (e.g. through accretion of knowledge) rather than by maturational change in underlying intellectual competence; and, more specifically, it will support the idea that acquisition of language is a contributor to this progression.

Method and Procedure

Training. A graded training procedure was used to introduce the "Daddy," "Mommy," "Baby" labels to the children. We used a family of stuffed teddy bears and a family of stuffed penguins in the training task. In the first phase the experimenter's set contained a large and a small penguin, while the child's set contained a large and a small bear. This meant that there was no object identity match yet, and the child was only confronted with two animals in each set. The experimenter explained the task to the child by saying "These bears and these penguins are each a family. In the your bear family, this (pointing to the larger bear) is the Daddy and this (pointing to the smaller bear) is the Mommy. In my penguin family this is the Daddy (pointing to the larger penguin) and this is the Mommy (pointing to the smaller penguin)."

The child was asked to repeat the labels. After the child could label all the stimuli in both sets the experimenter asked "If I put my sticker under my Daddy (Mommy) penguin, your sticker is under your Daddy (Mommy). Look, my sticker is under my Daddy; where do you thing your sticker is?" The child was then allowed to search for the sticker. Phase 2 was identical to Phase 1 except for the addition of a small bear and a small penguin resulting in two families consisting of three animals to which the labels "Daddy," "Mommy," and "Baby" were applied.

---

3. An alternative would have been to use "Big," "Medium," and "Little," however, young children are often quite slow to acquire relational terms such as "big" and "little," "high" and "low," etc. and they are often applied attributionally before they are applied relationally (Donaldson & Wales, 1970; Smith, Rattermann and Sera, 1988).

In Phases 3 and 4 we introduced competing object identity choices. That is, we tried to created the same tension that the children would face later in the mapping task. To do this, we gave both the experimenter and the child families of penguins. The sizes of the penguins were designed to create a cross-mapping between the stimuli in the experimenter's set and the child's set (e.g., the experimenter's family might contain sizes 1,2, and 3 and the child's family might contain sizes 2, 3, and 4). In phase 3 only two penguins were used in each family, while in phase 4 there were three penguins in each set. (See Figure 2.) Throughout the training task the child labeled both the experimenter's objects and the child's objects after every other trial.

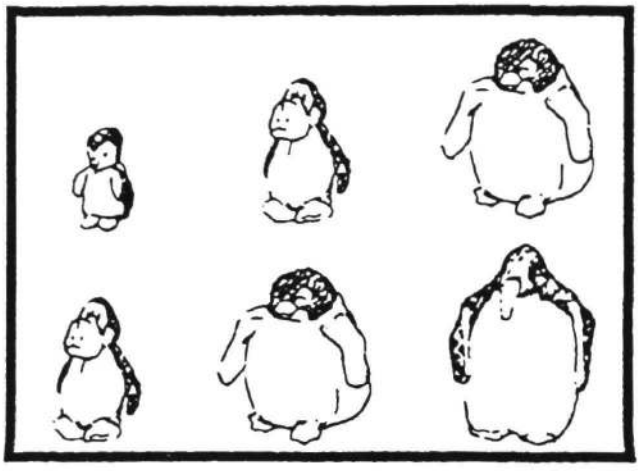


Figure 2. Stimulus set used in Phase 4 of training.

Mapping Task. After the training, each child was tested using the perceptual-mapping task using the stimuli described above (See Figure 1). Both the rich and the sparse stimuli were used, with half the children being tested with the sparse stimuli then the rich and the other half tested in the opposite order. Each child performed 28 trials; 14 sparse trials and 14 rich trials. The family labels were used in the same manner as in the training task.

Results

As can be seen in Figure 3, the children's performance in the labeled conditions was significantly higher than their performance when labels were not used with both the sparse stimuli (t (34) = 4.792, $p<.001$) and the rich stimuli ($t(34)=5.423$, $p<.001$). The use of relational labels helped the three-year-olds truly respond relationally even in the face of a very tempting object choice.

There was also a small effect of object richness in that the few mistakes the children made in this study were made in the rich object condition. A 2 (Order of stimulus type) x 2 (Random

order) x 2 (Object complexity) analysis of variance confirmed a significant effect of Object complexity F (1,20) = 4.44 p < .047.

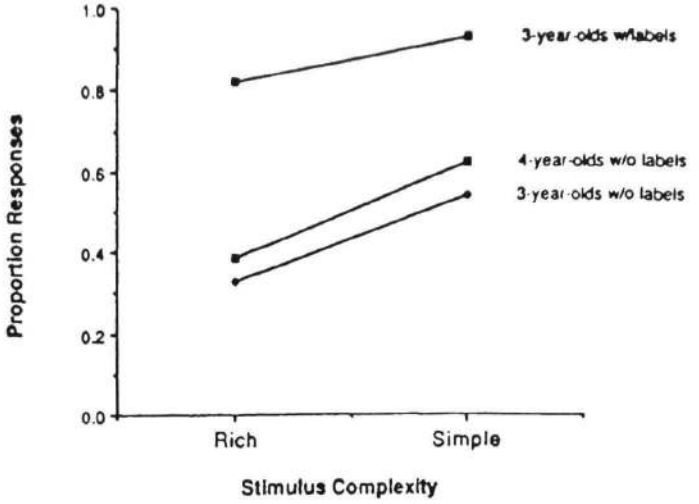


Figure 3. Results of labeling and non-labeling tasks.

Simulation

The use of relational labels helped the young children in our task to respond relationally. In fact, the familiar relational labels allowed the three-year-olds in this experiment to surpass the performance of the four-year-olds in the original mapping task suggesting a role of language in the perception of similarity.

More generally, this improvement in children's performance with relational labels strengthens the case for the domain-knowledge account of the development of similarity. That is, it suggests that children's model of processing is the same as that of adults. To further test this hypothesis, we carried out a computer simulation of the performance of children and adults in this task (Gentner, Markman, Rattermann & Kotovsky, 1990; Rattermann & Gentner, 1990). We gave propositional representations of the stimuli used in these experiments to the Structure-mapping Engine (SME) (Falkenhainer, Forbus & Gentner, 1986; 1989). (See Figure 4.) Based on the hypothesis that the accretion of domain knowledge is driving changes in similarity use, we formed two different knowledge representation of the stimulus sets.

We begin by making several working assumptions. We assume that children can vary in the degree of higher-order relations[4] present in their representation of the stimuli. We further assume that one role of language is to make the relational structure salient and increase the probability that the higher-order relations will be represented. Finally, we assume that children, in the absence of relational labels, possess shallow representations of the stimuli consisting of object attributes and first-order relations (The portion of Figure 4 in the dashed box.). When labels are provided they aid the children in forming

4. First-order relations are relations between objects, object-attributes or functions. Higher-order relations are relations between relations.

a systematic representation containing object attributes and an elaborated higher-order relational structure. Specifically, we assume that the higher-order relation of steady change in size is more likely to be represented when relational labels are used. In order to mimic the simple and the rich stimulus sets, we varied the number of object attributes; the rich objects possessed five attributes and the simple objects possessed three attributes. Given the systematic representations of the stimulus sets, SME's preferred mapping[5] was based on relational similarity for both the rich and the simple stimulus sets. Given shallow representations, however, SME's preferred mapping was based on relational similarity with the simple stimulus sets but based on object similarity with the rich stimulus sets. These results mimic our findings with the developmental task.

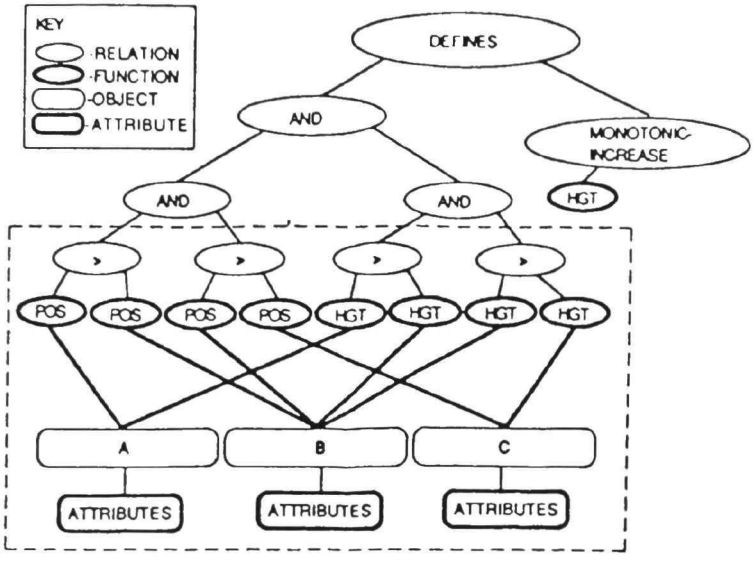


Figure 4. Knowledge representation used in simulation of developmental results.

Conclusions

In conclusion, this research suggests that the use of familiar relational labels can improve children's ability to perform analogical mappings. There is support, both empirical and computational, for the conjecture that children and adults may use the same type of similarity processes and that it is changes in domain representations rather than changes in cognitive competence that cause the observed developmental improvement and (2) the acquisition and use of language - specifically relational language - is an important contributor to this development.

References

Brown, A. L. (1989). Analogical learning and transfer: What develops? In S. Vosniadou & A. Ortony (Eds.), Similarity and analogical reasoning (pp. 369-412). London: Cambridge University Press.

Carey, S. (1984). Are children fundamentally different kinds of thinkers and learners than adults? In S. Chipman, J. W.

5. SME performs local matches and by taking advantage of connectivity produces structurally consistent interpretations of analogies. It then performs structural evaluations of each interpretation. We based our results on SME's preferred evaluations.

Segal, & R. Glaser (Eds.), Thinking and learning skills: Current research and open question (Vol. 2, pp. 485-517). Hillsdale, NJ: Erlbaum.

Crisafi, M. A., & Brown, A. L. (1986). Analogical transfer in very young children: Combining two separately learned solutions to reach a goal. Child Development, 57, 953-968.

Donaldson, M., & Wales, R. (1970). On the acquistion of some relational terms. In J. R. Hayes (Ed.), Cognition and the development of language (pp. 269-278). New York: Wiley.

Falkenhainer, B., Forbus, K. D. & Gentner, D. (1986). The structure-mapping engine (Tech. Rep. No. UIUCDC-R86-1275). Urbana, IL: University of Illinois, Department of Computer Science. Also appears in Proceedings of the Fifth National Conference on Artificial Intelligence (pp.272-277), Philadelphia, PA: Morgan Kaufmann.

Falkenhainer, B., Forbus, K., & Gentner, D. (1989/90). The structure-mapping engine: Algorithm and examples. Artificial Intelligence, 41, 1-63.

Gelman, S. A., & Markman, E. M. (1987). Young children's inductions from natural kinds: The role of categories and appearances. Child Development, 58,1532-1541.

Gentner, D. (1988). Metaphor as structure mapping: The relational shift. Child Development, 59, 47-59.

Gentner, D., Markman, A., Rattermann, M.J., & Kotovsky, L. (1990) Similarity is like analogy. Paper presented at the Midwest Artificial Intelligence and Cognitive Science Conference, Carbondale, Illinois.

Gentner D., & Toupin, C. (1986). Systematicity and surface similarity in the development of analogy. Cognitive Science, 10, 277-300.

Piaget, J., Montangero, J., & Billeter, J. (1977). Les correlats. In J. Piaget (Ed.), L'Abstraction reflechissante. Paris: Presses Universitaires de France.

Rattermann, M.J., & Gentner, D. (1990). The development of similarity use: It's what you know, not how you know it. Paper presented at the Midwest Artificial Intelligence and Cognitive Science Conference, Carbondale, Illinois.

Rattermann, M. J., Gentner, D., & DeLoache, J. (1989). Effects of competing surface similarity on children's performance in an analogical task. Poster presented at the biennial Meeting of the Society for Reasearch in Child Development, Kansas City, Missouri.

Smith, L. B., Rattermann, M. J., & Sera, M. (1988). "Higher" and "lower": Comparative and categorical interpretations by children. Cognitive Development, 3, 341-358.

Tversky, A. (1977). Features of similarity. Psychological Review, 84, 327-352.

Waxman, S. & Gelman, R. (1986). Preschoolers use of superordinate relations in classification and language. Cognitive Development, 1, 139-156.

Representational Issues in Analogical Transfer

Colleen M. Seifert Kenneth C. Gray

Department of Psychology
University of Michigan

Abstract

Lack of transfer may result in part from a critical, though often ignored factor: the form of the initial representation of information during the process of analogical transfer. Using a Gick and Holyoak (1980, 1983) replication, in which subjects read a story in the guise of a memory experiment, subjects were later required to solve a problem which could be solved using an analogous strategy suggested by the story. Transfer performance was measured by the presence or absence of this target solution in subjects' protocols. The text of the original General story (from Gick & Holyoak) was modified slightly in one condition, where one role in the story was replaced by another type of actor. The changes were minor, as shown by the fact that the story modification did not affect similarity ratings between the story and problem. However the changes did appear to affect subjects' initial representation of the story and, as a result, improve subsequent transfer to the problem. The results indicate that forming an initial representation of the story that is congruent with important features of the problem is critical for analogical transfer. Subjects' abstraction of a general problem solving schema is an inadequate explanation of these results.

Introduction

Contrary to naturally occurring examples of analogical reminding, many psychological experiments have demonstrated that people have a difficult time remembering and utilizing prior examples that are only abstractly related to the current situation (Holyoak & Nisbett, 1988). Despite all the attention lack of analogical transfer has received, key factors remain to be addressed about the transfer problem. In this paper, we argue that the case against analogical reminding is limited by a failure to adequately take into account subjects' initial memory representation of presented material.

Analogical transfer is important to study because of its obvious significance in learning and problem solving (Ross, 1987). In addition, however, we believe that investigating analogy is an appropriate method to gain insight about the representation of information in memory. Retrieval has often been assumed to be an automatic process that is dependent solely on matching an input to the contents of memory (Holyoak & Nisbett, 1988). Our basic contention is that such a "simple memory" model of episode retrieval will not account for the examples of reminding that do occur in the world. This "simple memory" model, which underlies many investigators' approaches to analogical transfer, involves the use of an overall similarity metric to identify the episode in memory with the most feature overlap (after Tversky, 1977). However, it appears that content feature matching alone is not sufficient to account for the richness of analogy observed in natural settings. Instead, we argue for a more complex model of memory-based analogy, where the determining factor in retrieval is the quality of the original encoding. For example, a great deal of inference is required to fully understand a story containing abstract relations as well as content features. Analogical transfer will not occur if the understander fails to perform elaborative inferences describing the relations between features in the example. Building an initial representation that contains both the abstract and content features is critical for any later analogical use based upon them. Thus, the ability to use analogies depends on efforts towards elaborative encoding of initial episodes. That encoding may be the key to analogy is supported by Gick and Holyoak (1983), where multiple exemplars in encoding produced better transfer rates, and by Seifert, McKoon, Abelson, and Ratcliff (1986), where abstract remindings occurred when subjects were given plenty of time to encode and summarize the initial stories.

How might representation play a role in lack of transfer? Consider Gick and Holyoak (1980, 1983), which concluded that people are unable to apply a general strategy learned in one situation to another. In one of their experiments, subjects read a story under the guise of a memory experiment. The story, titled "The General", describes a general's exploits in overthrowing a dictator (see Table

1). Following the story, subjects were given a problem to solve, Duncker's (1945) ray problem (also included in Table 1).

Table 1

The General

A small country was ruled from a strong fortress by a dictator. The fortress was situated in the middle of the country, surrounded by farms and villages. Many roads led to the fortress through the countryside. A rebel general vowed to capture the fortress. The general knew that an attack by his entire army would capture the fortress. He gathered his army at the head of one of the roads, ready to launch a full-scale direct attack. However, the general then learned that the dictator had planted mines on each of the roads. The mines were set so that small bodies of men could pass over them safely, since the dictator needed to move his troops and workers to and from the fortress. However, any large force would detonate the mines. Not only would this blow up the road, but it would also destroy many neighboring villages. It therefore seemed impossible to capture the fortress. However, the general devised a simple plan. He divided his army into small groups and dispatched each group to the head of a different road. When all was ready he gave the signal and each group marched down a different road. Each group arrived together at the fortress at the same time. In this way, the general captured the fortress and overthrew the dictator.

The Ray Problem

Suppose you are a doctor faced with a patient who has a malignant tumor in his stomach. It is impossible to operate on the patient, but unless the tumor is destroyed the patient will die. There is a kind of ray that can be used to destroy the tumor. If the rays reach the tumor all at once at a sufficiently high intensity, the tumor will be destroyed. Unfortunately, at this intensity the healthy tissue that the rays pass through on the way to the tumor will also be destroyed. At lower intensities the rays are harmless to healthy tissue, but they will not affect the tumor either. What type of procedure might be used to destroy the tumor with the rays, and at the same time avoid destroying the healthy tissue?

General Story and Ray/Tumor Problem from Gick and Holyoak (1980, 1983).

The plan used by the general to capture the fortress may be adapted into an analogous solution to the ray problem. The doctor can direct several low intensity rays from different sources to converge on the tumor. Gick and Holyoak's results were that only thirty percent of

subjects applied the strategy from The General story to the ray problem. Even when told to use the same solution, only some of the subjects (75%) were able to apply it correctly. According to Gick and Holyoak (1983), analogical transfer depends on subjects' ability to abstract a "convergence schema" from the story and problem. The schema, as proposed by Gick and Holyoak, describes problem types for which the convergence solution is an appropriate plan. It contains commonalities between separate episodes only in terms of problem solving actions and states.

In our view, the convergence schema cannot be the sole determinant of transfer. In order to be reminded of the prior story, one must have encoded that story with a similar set of dominating features. Subjects' original encoding may not have included the particular inferences necessary to generate the connection between the stories. Of course, it may be possible upon reflection to identify an analogous relationship; however, the critical question in spontaneous analogy is not whether you can generate such a link given the two cases, but whether each case individually sets up a memory representation such that they are likely to be similarly encoded into memory.

From this perspective, it becomes clear that an important factor is how each episode is structured for presentation, so that the dominant features one expects to be encoded a priori are in fact the ones encoded by subjects. In the present experiment, we attempted to manipulate the representation formed for the story to affect the rate of transfer to the problem.

The present experiment is, in part, a replication of Gick and Holyoak's (1980, 1983) transfer experiments. Subjects are presented with a story which introduces a solution to a problem. Later, they are given an analog problem in a different domain to solve. One condition included the story and problem as in Gick and Holyoak (1980), as shown in Table 1. In a second condition minor modifications were made to the story in an effort to alter subjects' initial representations of the story to facilitate transfer to the problem. Specifically, the problem suggests the need to destroy an "enemy within" the body-- the tumor. However, this role is more difficult to observe in The General story. In the original story, the presence of the dictator was the status quo. As far as the reader

can tell, the dictator had always controlled the fortress. A representation of this point may be quite different from that for the tumor in the analogous portion of the ray problem. A tumor is an object that has appeared (possibly suddenly) in the body. It would not be represented as the status quo. Rather, the appearance of a tumor is more like a sudden invasion of some foreign agent. The new version of The General story, shown in Table 2, retells the story, simply replacing the dictator from the original version with "terrorists".

Table 2

The General- Terrorist version

A small country was ruled from a strong fortress by a dictator. The fortress was situated in the middle of the country, surrounded by farms and villages. Many roads led to the fortress through the countryside. **A small group of terrorists had taken over and barricaded themselves in the fortress. An army** general vowed to capture the fortress. The general knew that an attack by his entire army would capture the fortress. He gathered his army at the head of one of the roads, ready to launch a full-scale direct attack. However, the general then learned that the **terrorists** had planted mines on each of the roads. The mines were set so that small bodies of men could pass over them safely, since the **terrorists** needed to move troops and workers to and from the fortress. However, any large force would detonate the mines. Not only would this blow up the road, but it would also destroy many neighboring villages. It therefore seemed impossible to capture the fortress. However, the general devised a simple plan. He divided his army into small groups and dispatched each group to the head of a different road. When all was ready he gave the signal and each group marched down a different road. Each group arrived together at the fortress at the same time. In this way, the general captured the fortress and overthrew the **terrorists.**

Revised General Story - terrorist version. Items in boldface were changed in this version from the original in Table 1.

By replacing the dictator with terrorists in the new version, we have highlighted this perspective, call it the "enemy within" perspective in The General story. When a memory representation for the terrorist version is set up, it should now reflect the "enemy within" perspective. The terrorist version thus highlights an additional commonality with the ray problem. Note that this new commonality is not part of the convergence schema.

Method

Subjects. Subjects were 36 University of Michigan undergraduates who participated for credit in a psychology course.

Materials. Duncker's (1945) ray problem and The General analog from Gick and Holyoak (1980, 1983) were used, along with The General- terrorist version. All materials are displayed in Tables 1 and 2.

Procedure. 21 subjects read the original version of The General and 15 subjects read the terrorist version for 3 minutes. They were then asked to write their recall of the story. After protocols were written, subjects attempted to solve the radiation problem. Following this, subjects were given a hint to "propose a solution suggested by the story." Finally, subjects were asked if they had seen the story or problem before in any context (and if so, discarded from the analysis).

Results

Table 1 shows the proportion of subjects who proposed the convergence solution to the ray problem after reading the original and the terrorist versions of The General. The left columns indicates the proportion who transferred the solution strategy spontaneously, without any hint to use the story. The second column gives the total proportion of subjects who transferred successfully after being told to use the story (this column includes the subjects from the first column). The last column gives the proportion of subjects who did not propose the convergence solution.

Table 1

	Before Hint	After Hint	No Transfer
Original Version	.19	.62	.38
Terrorist Version	.40	.60	.40

A chi-square test for association revealed that the proportion of subjects who transferred before the hint was significantly greater in the terrorist story condition ($\chi^2(2) = 3.94$, $p < .025$).

Discussion

The modifications in the General story, though minor in amount of textual change, were successful in increasing the rate of transfer. The point made by this manipulation is more subtle than saying that more similar stories result in better transfer; rated similarity when given both analogues is the same in the original and the changed versions. An independent group of 19 subjects given either the original version and the ray problem, or the terrorist version and the ray problem were asked to rate the similarity of problem to story on a scale from 0 - 10, where 0 was labelled "not at all similar" and 10 was labelled "extremely similar." No explicit instructions regarding similarity judgments were given. The mean ratings given were 7.9 for the original version and 7.6 for the terrorist version. This difference is not significant ($t(17) = 0.21$, $p > .8$).

Therefore, the changes in the terrorist version did not result in a "more similar" judgment when story and problem are compared. Instead, the representation formed when reading the changed story resulted in better analogical access and transfer when tested on the ray problem. The critical point to be made here is that the features likely to be used at encoding will dominate any use of the episode in analogical processing. Therefore, care must be taken to determine the nature of the representation built for each single presentation of each example, rather than the perceived similarity during comparison.

The ability to be reminded based on abstract features requires encoding both episodes with similar features. Because the analogues used in experiments require a fairly sophisticated representational system to characterize the target similarities, care must be taken to ensure that the representation subjects take away from their presentation must be ones that are candidates for transfer. Because of the dependence on materials, and in particular the use of a small set of classic examples for replications and extensions, conclusions are dependent on ensuring that the materials

satisfy the above constraints. When they do, they provide a methodology for examining the features people encode about the world that do lead to transfer to new problem domains.

Acknowledgements

This research was sponsored by a contract between The University of Chicago and The University of Michigan under ONR contract No. N00014-88-K-0295. The authors wish to thank Kristian Hammond and Susan Chipman.

References

Duncker, K. (1945). On problem solving. *Psychological Monographs*, 58(270).

Gick, M., & Holyoak, K. (1980). Analogical problem solving. *Cognitive Psychology*, 12, 306-355.

Gick, M., & Holyoak, K. (1983). Schema induction and analogical transfer. *Cognitive Psychology*, 15.

Holyoak, K. & Nisbett, R. (1988). Induction. In E. Smith and R. Sternberg (Eds.) *The Psychology of Human Thought* (pp.50-91). Cambridge: Cambridge University Press.

Ross, B. H. (1987). This is like that: The use of earlier problems and the separation of similarity effects. *Journal of Experimental Psychology: Learning, Memory and Cognition*, 13(4), 629-639.

Seifert, C. M., McKoon, G., Abelson, R. P., & Ratcliff, R. (1986). Memory connections between thematically similar episodes. *Journal of Experimental Psychology: Human Learning and Memory*, 12 (2), 220-231.

Tversky, A. (1977). Features of similarity. *Psychological Review*, 84, 327-352.

Analogical Mapping During Similarity Judgments*

Arthur B. Markman
University of Illinois
Department of Psychology
Beckman Institute

Dedre Gentner
University of Illinois
Department of Psychology
Beckman Institute

Abstract

We propose that carrying out a similarity comparison of two objects or scenes requires that their components be aligned in a manner akin to analogical mapping. We present an experiment which supports this claim and then examine a computer simulation of these results which is consistent with the idea that a process of mapping and alignment occurs during similarity judgments.

Introduction

A process for calculating the similarity of two things has been assumed by nearly every model of categorization (Medin and Shaffer, 1978, Smith and Medin, 1981) and problem solving (Ross, 1984, Gick and Holyoak, 1985). A better comprehension of the processes which govern similarity would result in a deeper understanding of the mechanisms which control many other cognitive processes.

The pioneering work of Tversky (1977) established that the similarity between two items is a function of the elements which the two items have in common (common features) and the elements possessed by one item but not by the other (distinctive features). In addition to outlining the importance of common and distinctive features, Tversky and his colleagues have set out a number of ways in which the salience of common and distinctive features can be determined (Tversky, 1977, Gati and Tversky, 1984).

Although the importance of common and distinctive features cannot be overstated, Tversky's work has not addressed the problem of how the representations of the objects are compared in order to determine which elements are common and which elements are distinctive. However, some recent work has begun to address this issue (Markman, Medin and Gentner, under revision, Goldstone, Gentner and Medin, 1989). This work suggests that object representations are aligned through a process akin to the mapping processes which have been proposed for analogical reasoning (Gentner, 1983, 1989, Holyoak and Thagard, 1989 and Hall, 1989).

In this paper we first outline how analogical mapping may be applicable to similarity. We then present an experiment designed to test whether mapping is a part of similarity judgments. Finally, we will examine the results of a computer simulation of this experiment.

Similarity and Structure Mapping

We will discuss the applicability of mapping to similarity with respect to Gentner's structure-mapping theory (SMT) (Gentner, 1983, 1989) specifically, but the general points are compatible with many current theories of analogical mapping (cf. Hall, 1989 for a review). According to SMT, the comparison of two scenes requires that their relational structures be

---

* This work was supported by Office of Naval Research contract N00014-89-J1272, National Science Foundation grant BNS 87-20301, and a University of Illinois Cognitive Science/Artificial Intelligence Fellowship. The authors would like to thank Ken Forbus, Douglas Medin, Janice Skorstad, Laura Kotovsky and Mary Jo Rattermann for their helpful comments on previous drafts of this manuscript. Thanks to Robert Parish for holding down the middle. A special thanks to Julie Hays for running subjects.

aligned. Thus, objects which play a common role in both scenes are likely to be placed in correspondence, while identical objects which play different roles in their respective scenes are unlikely to be placed in correspondence. In addition, Gentner's systematicity and structural consistency principles reflect subjects' tendency to preserve relational structure (e.g. causal relations, goals, or other higher order relations) even when the objects themselves are dissimilar.

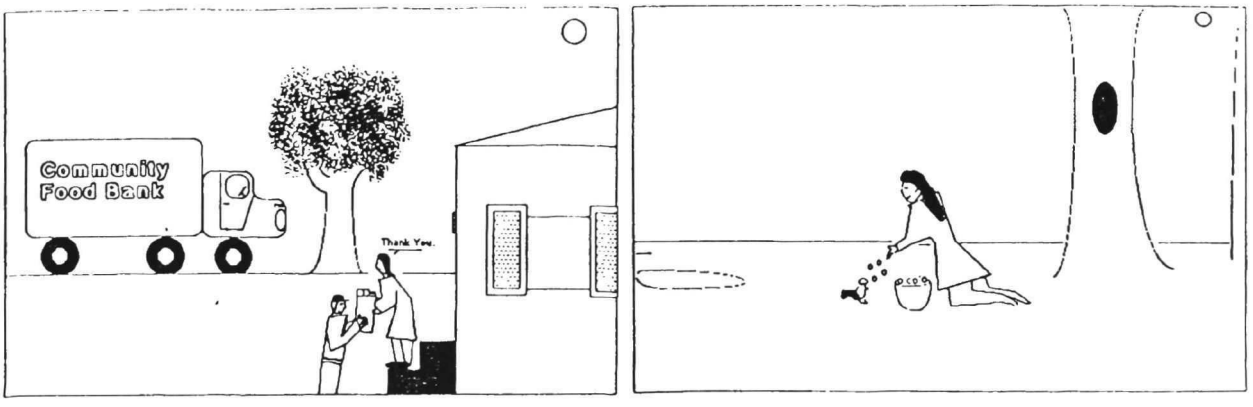
Recent research has shown that similarity judgments are sensitive to the relational structure of the stimuli (Markman, Medin and Gentner, submitted, Goldstone, Medin and Gentner, 1989, Rattermann and Gentner, 1987). For example, Rattermann and Gentner found that subjects considered pairs of stories that had similar relational structure and different characters to be more similar than pairs that had similar characters and a different relational structure. However, sensitivity to relations yields only indirect evidence that an analogical mapping process takes place during similarity judgments. The following experiment will address this question more directly.

Experiment 1

This experiment was designed to test the claim that an analogical mapping process takes place during similarity judgments. Subjects were shown a base scene and a target scene (see Figure 1). One of the objects in the base scene was highlighted and the subject was asked which objects 'goes with' that object in the target scene. We call this task one-shot mapping (1map).

In each pair of scenes one object was *cross-mapped*. Gentner and Toupin (1986) define a cross-mapping as a comparison in which there are perceptually similar objects in two scenes, but the perceptually similar objects play different roles in the relational structure of each scene. For example, in Figure 1, the two women are highly similar perceptually. However, in the top scene the woman is receiving food while in the bottom scene she is giving food away. Thus, the perceptually similar objects play different roles in their respective scenes.

Using this tension between perceptual similarity and relational similarity, we examined subjects preferences in three conditions. The first group of subjects (1map), was shown the pair of scenes. The experimenter pointed to the cross-mapped and asked the subject to point to the object in the target which 'went with' the cross-mapped object. The second group of subjects (3map) was asked to give the objects that 'went with' three of the significant objects in the target scene including the cross-mapped object. Finally, the third group of subjects (sim->1map) was asked to rate the similarity of the two scenes first, and then they were asked to point to the object that 'went with' the cross-mapped object.



(a) (b)

Figure 1. Sample stimulus in Experiment 1a.

We predicted that the subjects in the 1map condition would tend to select the nearly identical object in the other scene. Since these subjects need only map a single object, they have no reason to consider the relational structure of the two scenes and thus will be guided by object similarity.

Furthermore, we predicted that subjects in the 3map condition who were asked to map three of the objects which play a role in the relational structure would tend to map objects based on the relational structure of the scenes. These subjects must consider how to make three consistent mappings. If the two cross-mapped objects are placed in correspondence, there is no justification for placing any of the other objects in correspondence. As a result, these subjects should be more likely to preserve the relational structure of the scenes, which would provide support for all three object mappings.

The key prediction, however, centers on the the sim->1map condition, where subjects first performed a similarity judgment and then perform a single mapping. We predicted that these subjects would also be likely to map objects based on the relational structure of the scenes. If, as we believe, similarity judgments require that the relational structures of the scenes be aligned, then subjects' mappings following a similarity judgment should reflect this relational structure.

Method

Stimuli. The set of stimulus pictures portrayed causal higher order relations (*causal HOR*). These pictures presented scenes with a goal structure or causal structure. For example, in the pictures shown in Figure 1 someone is giving food to someone else. Each of these pictures had a cross-mapping as well.

Procedure. Subjects were run one at a time. They were seated at a table with an experimenter seated behind them. Subjects participated in only one experimental condition. The experimenter had no knowledge of the hypothesis being tested.

Subjects in the one-shot mapping (1map) condition were shown each of the base/target pairs in turn. The experimenter pointed to the cross-mapped object and asked the subject to point to the object in the other picture that went with that object. The subject's response was recorded and the next pair of pictures was presented. After completing the mapping task, subjects rated the similarity of each pair on a scale from 1 to 9.

Subjects in the three mappings (3map) condition were also shown each of the base/target pairs. However, in this case, the experimenter pointed (one at a time) to each of the three objects which made up the central relational structure of the scene. Subjects were asked to point to the objects which went with each of those objects. The cross-mapped object was always tested first. After completing the mapping task, these subjects were also asked to rate the similarity of all of the pairs of pictures.

Subjects in the similarity first then one-shot mapping task (sim->1map) first rated the similarity of a pair of pictures and then were shown the cross-mapped object and asked which object in the other picture went with it.

Finally, a control condition was run. It could be argued that any effects found in the sim->1map condition arise because subjects have greater exposure to the pictures in this condition before performing the one-shot mapping. In order to control for this possibility, one final group of subjects was shown the set of pictures one at a time and told to study them carefully for a later memory experiment. The subject saw each picture for five seconds, roughly the amount of time subjects see the pictures while making similarity comparisons. After examining the entire set of pictures, subjects in the control condition performed the 1-shot mapping task.

Design. There were 4 (3 mapping conditions and control) between subjects conditions in this design. Order of stimulus presentation was counterbalanced.

Subjects. Subjects were 24 undergraduates from the University of Illinois who received course credit for their participation in a single 15 minute session. Subjects were randomly assigned to one of six experimental conditions.

Results. Subjects' responses were recorded as being either object-based mappings, relation-based mappings or spurious mappings. Object-based mappings were those choices which placed the cross-mapped objects in correspondence. Relation-based mappings were those choices which preserved the relational structure of the scenes. Any other choice was considered a spurious mapping. The number of spurious mappings was less than 1% of the total number of responses and will not be considered further here. The proportion of relational responses on the cross-mapped object for each subject is shown in Figure 2.

As predicted, subjects who performed a similarity judgment prior to mapping made more relational responses than subjects who did not perform a similarity judgment prior to mapping. A one-way analysis of variance on the three experimental conditions indicates that, as predicted, there is a significant effect of mapping condition ($F(2.21)=3.76$, $p<.05$). A planned comparison indicates that significantly more relational responses were given in the sim->1map condition than in the 1map condition ($F(1,21)=7.51$, $p<.01$ one-tailed). More relational responses were given in the 3map condition than in the 1map condition, but this difference was only marginally significant ($F(1,21)=2.233$, $.05<p<.1$, one tailed). Finally, there was significantly more responding in the sim->1map condition than in the control condition ($t(14)=2.326$, $p<.05$ one tailed).

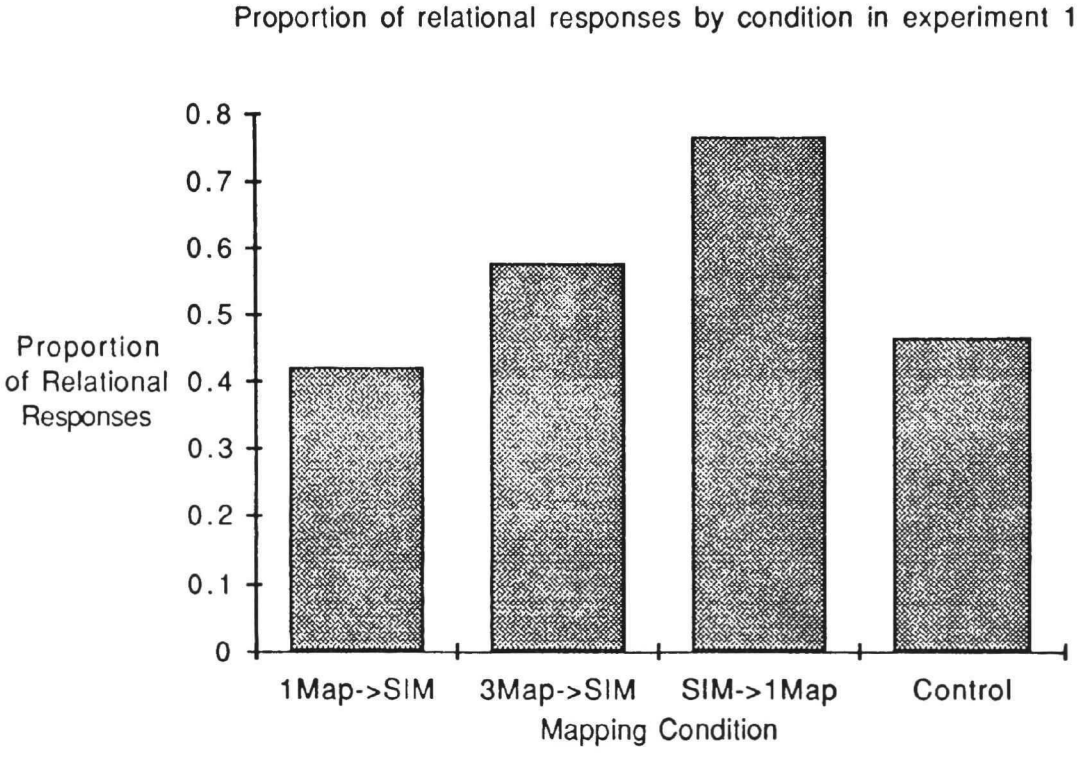


Figure 2. Graph of proportion of relational responses made by subjects in each of the mapping conditions in Experiment 1.

A correlation was done between the mean similarity judgments for each stimulus pair in each condition and the total number of relational responses given to that stimulus pair. This correlation was performed to ensure that subjects in the sim->1map condition did not simply

find the stimulus pairs more similar than subjects in other conditions and thus gave more relational responses to stimuli in this condition. As expected, the correlation between similarity and number of relational responses across all mapping conditions is not significant ($r(22)=.35$, $p>.05$). Separate correlations between mean similarity and number of relational responses were also determined for each mapping condition. The correlation in the sim->1map condition were significant. ($r(6)=.64$ $p<.05$ one tailed). The correlations were not significant for any of the other conditions.

Discussion

As predicted, subjects in the sim->1map condition made more relational responses than subjects in either the 1map or control conditions. This means that, given a task that ostensibly requires object alignment, subjects prefer to preserve the relations between objects. These results support the claim that an alignment process similar to analogical mapping takes place during similarity judgments.

Furthermore, the correlation between number of relational responses given to an item and the mean similarity rating is only significant in the sim->1map condition. Since prior research indicates that similarity judgments are highly sensitive to relational structure (Markman, Medin and Gentner, under revision, Goldstone, Medin and Gentner, 1989), we interpret this correlation to indicate that, when subjects were aware of the common relational structure, they placed objects in correspondence based on their role within the relational structure. Furthermore, the non significant correlation in the 1map case indicates that, although subjects may have chosen their mapping based on the similarity between the objects, their global similarity judgments were still based on the relational similarity of the scenes.

A simulation using the Structure Mapping Engine (SME)

A computer simulation of the mapping process for the causal HOR stimulus was performed using the structure mapping engine (SME) program which was designed to implement the structure mapping theory of analogical mapping (Falkenhainer, Forbus and Gentner, 1986, 1989). The stimulus pictured in Figure 1 was encoded into a propositional representation used by SME. The representation of the scene in Figure 1a is depicted graphically in Figure 3.[1] In order to capture the cross-mapping the cross-mapped objects were given identical descriptions including a number of shared attributes, but the objects played different roles in their respective relational structures.

SME generates all possible interpretations of the match between two scenes. SME begins by proposing local matches between identical predicates in the base and target. These local matches are then connected into larger mappings provided that these matches fit into a consistent relational structure. Next, these larger matches are combined together into maximal sets with the proviso that they are structurally consistent. These maximal sets, called GMAPs, are then evaluated for their systematicity (see Falkenhainer, et. al. 1989 and Forbus and Gentner, 1989 for a discussion of the evaluation procedure). For these simulations, SME was configured using literal similarity rules which allow the system to map both attributes and relations. This configuration allows both object similarity and relational similarity to play a role in the generation of GMAPS.

For the causal HOR stimulus pair, the relational interpretation was clearly preferred. The GMAP with the highest evaluation score placed the woman receiving food and the squirrel receiving food in correspondence, as well as the man giving food and the woman giving food. The

---

[1]The representation of the causal HOR in Figure 4a is slightly simplified. All of the objects shown in the scene were included in the representation and various relations were placed between these stimuli. In order to conserve space, only two of those objects were included in the figure.

mapping preferred by SME is the same mapping preferred by subjects in the sim->1map condition where 7/8 (88%) of the subjects chose the relational interpretation.

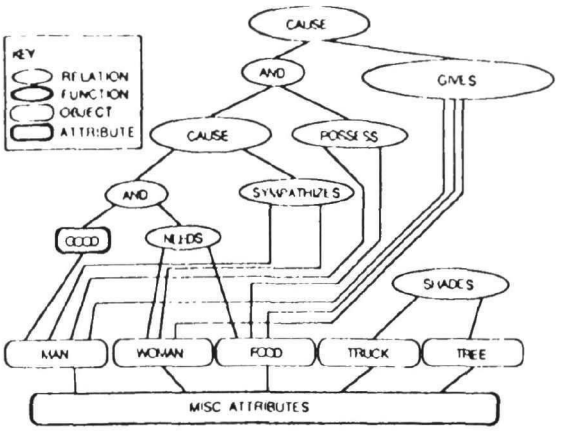


Figure 3. Propositional representation of scene in Figure 1a. This representation was used in the SME simulation of Experiment 1.

SME was also able to generate the object match interpretation of the scenes. This GMAP was given a lower evaluation score than the relational interpretation, but was given a higher evaluation score than any of the other interpretations. The simulation results are consistent with subjects' data here as well. Recall that subjects rarely (less than 1% of all trials) gave a response other than an object match or a relational match. Like human subjects, SME did not generate any highly-rated GMAPS corresponding to a spurious match.

Conclusion

The results of the experiment performed here support the claim that, in order to compute similarity, subjects must perform a relational mapping to achieve a structural alignment between the two scenes. This pattern of results is consistent with an SME simulation of the data. Thus, structure-mapping theory predicts that subjects will place objects in correspondence based on their position within the representational structure. The consistency of the predictions of structure-mapping theory with the data obtained from this similarity study demonstrate the degree to which the processes of analogical mapping and similarity judgments are alike.

References

Falkenhainer, B, Forbus, K. D., and Gentner, D. (1989). The Structure Mapping Engine: Algorithm and Examples. Artificial Intelligence. 41(1), 1-63.

Forbus, K. D. and Gentner, D. (1989). Structural Evaluation of Analogies: What Counts? in the proceedings of the 11th Annual Conference of the Cognitive Science Society. Ann Arbor, MI.

Gati, I. and Tversky, A. (1984). Weighting Common and Distinctive Features in Perceptual and Conceptual Judgments. Cognitive Psychology. 16, 341-370.

Gentner, D. (1983). Structure Mapping: A theoretical framework for analogy. Cognitive Science. 7, 155-170.

Gentner, D. and Toupin, C. (1986). Systematicity and Surface Similarity in the Development of Analogy. Cognitive Science. 10, 277-300.

Gick, M. L. and Holyoak, K. J. (1983). Schema Induction and Analogical Transfer. Cognitive Psychology. 15, 1-38.

Goldstone, R. L., Medin, D. L., and Gentner, D. (1989). Relations Relating Relations. in the proceedings of the11th Annual Conference of the Cognitive Science Society. Ann Arbor, MI.

Hall, R. P. (1989). Computational Approaches to Analogical Reasoning: A comparative analysis. Artificial Intelligence. 39, 39-120.

Holyoak, K. J. and Thagard, P. (1989). Analogical Mapping by Constraint Satisfaction. Cognitive Science. 13, 295-355.

Markman, A. B., Medin, D. L., and Gentner, D. (in revision). An analysis of the Perceptual-Verbal Distinction in Similarity Judgments.

Ross, B. (1984). Remindings and their Effects in Learning a Cognitive Skill. Cognitive Psychology. 16, 371-416.

Smith, E. and Medin, D. L. (1981). Concepts and Categories. Cambridge, MA: Harvard University Press.

Tversky, A. (1977). Features of Similarity. Psychological Review. 84(4), 327-352.

Goal Similarity in Analogical Problem Solving

Bruce D. Burns

University of California, Los Angeles

Abstract

The role of goal similarity in analogical problem solving was investigated using highly simplified chess positions. Goal similarity was manipulated by instructing subjects to make an attacking or defensive move. Subjects received training positions, followed by a set of testings positions, each solvable by mapping to a training position. A normal chess position was also given. Testing positions maximally similar to training positions (including similarity of goals) were responded to most quickly, though this effect was not found for all positions. It was also found that when subjects had to avoid a fatal threat in a normal chess position, they were more likely to successfully defend against that move if they were told that they were losing than if told they were winning. The results indicate that goal similarity influences analogical problem solving.

Introduction

Seeing a new problem as analogous to an old problem can help develop a solution (Brown, Kane, & Echols, 1986; Carbonell, 1983; Gick & Holyoak, 1980, 1983). To do this requires retrieval of a source analogue and construction of a mapping between the source and the target analogue. How these analogical processes can be achieved remains uncertain. The goals of a cognitive system have often been proposed as important factors in analogy mapping, because the purpose of analogy mapping in problem solving is to accomplish the goals of the problem solving. This has led many artificial intelligence theorists to argue for the importance of goal accomplishment to retrieval of analogies (e.g., Carbonell, 1983, 1986; Hammond, 1989; Schank, 1982; Winston, 1980, 1982), as well as more psychologically orientated theorists (e.g., Holyoak and Thagard, 1989; Gentner, 1989) to argue that goal similarity has a role in analogical mapping.

This research was supported by Contract MDA 903-89-K-0179 from the Army Research Institute. I wish to thank Ramin Gabizadeh for data collection and Keith Holyoak for extensive comments. Correspondence should be sent to Bruce Burns, Department of Psychology, UCLA, Los Angeles, CA 90024, USA; or e-mail burns@cognet.ucla.edu

For the concept of goal similarity to be meaningful it needs to be distinguished from the more general influences of semantic similarity and structural consistency. The degree of semantic similarity relates to the similarity of the elements of a source and target analogy. Structural consistency refers to the degree to which the elements play similar roles in the analogues, which can be equated with a simple criterion for mapping: If two propositions are mapped, then their constituent predicates and arguments should also be mapped. There is a reasonable amount of psychological evidence that semantic similarity (e.g., Holyoak & Koh, 1987; Ross, 1987) and structural consistency (e.g., Gentner & Toupin, 1986; Holyoak & Koh, 1987; Ross, 1987, 1989) influence analogical mapping, although there is disagreement on exactly how these factors affect mapping. How, or whether, goal similarity is distinct from these factors is disputed. One reason for this dispute is that, despite the popularity of goal similarity in theories of analogical mapping and problem solving, there is very little psychological evidence that it plays a role. One suggestive study was that of Brown et al. (1986). They found that children who were directed to focus on the goal structure of a problem were better able to transfer solutions to analogous problems than were children not so directed. Despite this, both groups were able to recall the goals when later tested, suggesting that their poor performance was not due to a simple memory failure. However, it is possible that the directions acted as a hint, thus aiding retrieval.

The present study aimed to find evidence that goal similarity plays a role in analogical problem solving, and to develop a task suitable for the further study of the its effects. If goal similarity has been a slippery concept, this is partly due to the lack of a operational definition that might be provided by a task that can demonstrate its effects.

The Task

The problem solving task used in this study is novel and therefore it is necessary to explain its nature and rationale in some detail. The task involved highly simplified chess positions. Chess was chosen largely because it is possible to clearly define goals in terms of attacking and defensive

moves. Different players given exactly the same position can have different goals (e.g., attack or defence) without requiring an overt change of the semantic or structural components of the problem.

The use of naive subjects dictated the use of a simple task. The positions consisted of two chess pieces on the board and one piece presented off the board, but waiting to be put on the board. For the attacking goal the task involved the subjects placing the piece that was off the board onto the board so as to guarantee that this piece would be able to capture an opposition piece on its next move. This next move occurs after one of those opposing pieces had had an opportunity to move. The task when the goal was defensive was the opposite of the attacking task: the subject must avoid the capture of a piece. The subject legally moved one of the white pieces already on the board in anticipation that the black piece off the board was about to be placed onto the board in an attempt to complete an attacking manoeuver similar to the one the subject used in the attacking task.

To help subjects achieve these goals they were trained on examples of two simple chess tactics known as *pins* and *forks*. These tactics are known by any experienced chess player, but they will not be familiar to novices. A fork manoeuver involves the attacking piece being placed so as to simultaneously attack two opposition pieces. Because the defending side's move consists of moving only one piece, only one piece will be able to escape the attack leaving the other to be captured. For a pin manoeuver the attacking piece is placed such that only one piece is directly attacked, but if this piece moves away then the one behind it can be captured. Hence a capture is still guaranteed. Examples of these training positions are presented in Figure 1 (an attacking bishop pin), Figure 2 (an example of a successful solution of the attacking bishop pin position), and Figure 3 (a successful solution to the defensive version of the bishop pin position).

After training on one of four attacking examples and on one of four defending examples, the subjects were given 40 testing trials consisting of similar simplified chess positions. Each testing position was a transformations of one of the eight training positions. Each transformation involved the same manoeuver (e.g., attacking bishop pin) but had the pieces changed (a change in semantic similarity), the relationship of the pieces to each other changed (structure inconsistency), or the goal changed (from attack to defence, or vice-versa). Because a subject received only two of the eight training positions (each involving a different manoeuver), transfer of a manoeuver was required when the subject was given one of the other training positions as a testing position. Therefore each testing position was classified by the type of transfer that would be required to get from the manoeuver used in one of the training position that the subject had been given, to the manoeuver in the training position of which the testing position was a transformation. The four transfer conditions were *similar*, *attack/defence*, *fork/pin*, and *bishop/rook*.



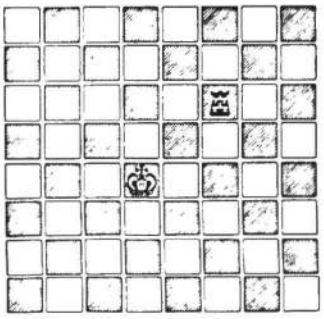


Figure 1. An attacking bishop pin position, as presented to a subject.

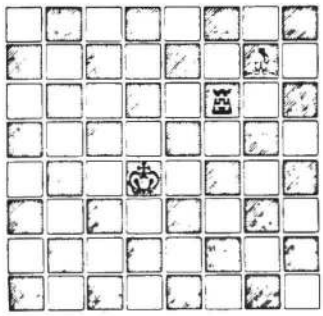


Figure 2. A solution to the attacking bishop pin position presented above.



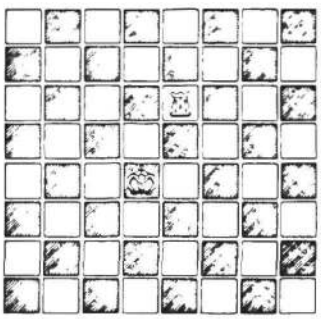


Figure 3. A solution to the defensive version of the bishop pin position presented above.

The similar condition involved the positions

that used the same manoeuver (e.g., attacking rook fork) as that used in one of the subject's training positions and hence they differed the least from the training position. Of course, a subject's training position given as a testing trial would fall into this condition, though it could no longer be considered an analogue because it would actually be identical to the training position. The attack/defence condition involved a testing position that kept the rook/bishop and pin/fork characteristics of the manoeuver constant, but swapped attack for defence or vice-versa. This condition is considered to involve a change of goal. The fork/pin condition kept the attack/defence and bishop/rook characteristics of the required manoeuver the same, but swapped fork for pin, or vice-versa. This is considered to represent a structural change of the manoeuver. The bishop/rook condition involves positions that maintain the attack/defence and fork/pin characteristics but swap between rooks and bishops as the attacking piece. This is considered to primarily be a semantic change of the manoeuver, but because rooks and bishops move differently some structural change is unavoidable.

Due to the transformation of the training position, some form of transfer would be required to map even to a testing position that was a transformations of one the testing positions that the subject had been given. To control this it will be possible to directly compare positions that were constructed using the same type of transformation, but represent different transfer conditions.

Note that over subjects, each testing position appeared in each transfer condition, so that which condition a testing position was part of depended solely on what the subject's training positions were. Response time was used as the measure of performance because most of these positions are simple enough to be eventually solved by exhaustive search.

To investigate the effect of goal similarity, the comparison of the similar and the attack/defence conditions is most critical. If goal similarity is important for mapping or retrieval, then subjects should perform better on the positions in the similar condition than they do in the attack/defence condition. If subjects find the attack/defence positions harder then this will provide evidence that changing the goal makes problem solving harder and hence support the claim that goal similarity affects the analogical mapping and/or retrieval. It could be argued that differences in difficulty could be due simply to the attacking and defending tasks being different; there might be no transfer because training for one goal does not help with the other. However, there is in fact an intimate link between these two tasks. To solve a defending position requires knowledge of the attacking move. Subjects could not be sure that their solution to a defending position is correct unless they know what the threatened attack is, hence training on the attacking manoeuver should help the subject solve the defensive version of the manoeuver. Successful training on the defensive manoeuver involves becoming aware of the attacking form of that manoeuver.

Better performance should also be found in the similar condition as compared to the fork/pin and bishop/rook conditions, as previous work on problem solving tasks provide ample evidence that structure and semantics affect analogical mapping. The finding of such differences is important, however, because it would lend support to the assumption that solving the task involves mapping to the training positions.

A further attempt to investigate the effects of goals involved attempting to manipulate goals at the time of solving a problem. In a final position, subjects were given a full chess position and instructed to try to make the best move they could as though it were a normal chess position. This position involved a fatal move which was either available to the subject or that the subject had to identify so that they could defend against it. To manipulate goals, subjects were told either that they were winning or losing. This information is in fact irrelevant to selection of a correct move, but may have altered their immediate goal in the position. Subjects who were told they are winning would be more likely to make aggressive moves than those told they were losing. If goals are important then it would be predicted that telling a subject that they are winning should help them find the correct attacking move in the appropriate position, while telling subjects that they are losing should assist them in finding the correct defensive move when it is required.

Method

Subjects

A total of 81 subjects participated in the experiment. All subjects were from an introductory psychology subject pool at the University of California, Los Angeles. Seven subjects were eliminated because they failed to complete the task in the available time and two were eliminated because they never understood the requirements of the task. This left a sample of 72 subjects (53 male, 19 female).

Materials and Apparatus

All positions and instructions were presented by an Apollo series 4000 workstation with a 19 in. color monitor. Subjects made their responses using a mouse with three buttons. Response times were measured to an accuracy of one second.

Training positions

There were three factors each with two levels (fork or pin; bishop or rook; attack or defence) that were crossed, yielding eight training positions. However, only four different training conditions were used as each player was given two training positions that were the exact opposite of each other. For example, if one of the training positions was a bishop, pin, attack, then the other position would be a rook, fork, defence. In this way the subject was exposed to all possible elements. In particular, they became familiar with both the attacking and the defending task. It was especially important that the subject got a chance to clarify the nature of the defending task, since it is a more complex task than the attacking task.

Several constraints were applied to the construction of the eight training positions. The first constraint was that every attacking position must not only have a valid attacking solution, but must also have a defensive solution, because the same position was given as a defensive task. In effect, this means that there were only four different basic positions. The defensive versions of the training positions simply involved changing the goals and the colors of the pieces. Second, an attacking position should allow a pin solution, or a fork solution. Third, because of the restriction that an attacking piece cannot be placed onto a square where it could be immediately captured, it is not possible to create a fork position where one of the defending pieces is the same as the attacking piece. Hence one defending piece was always the opposite (in terms of rook and bishop) of the attacking piece and the other was a king. The use of a king made it easier to fulfill some of the other constraints. A fourth constraint was that it had to be taken into account that the training positions would be used as the basis for the construction of testing positions. The training positions therefore needed to be transformable in the appropriate ways.

Testing positions

Each testing position was based on one of the four basic training positions, but transformed in various ways. A total of 10 transformations were applied to each of the four basic training positions, yielding a total of 40 testing positions. Five of these transformations (*identical, rotation, change-piece, change-structure,* and *queen*) appeared for every training position. An identical transformation was the training position itself, which never appeared until at least the 22nd trial, by which time the subject had been exposed to all other possible manoeuvers. The rotation transformation was the training position rotated. The change-piece transformation was the same as the training position except that the defending piece that wasn't a king was replaced by a knight. The change-structure transformation replaced a defending piece and changed the relative position of the defenders (but an attacking maneuver of a similar type was still possible). The queen transformation used a queen as the attacking piece but allowed a similar attacking manoeuver to that used in the transformed training position. For fork training positions, five other types of more complicated transformations were used, but for pins (which are less flexible) these five were repeated so that each pin transformation was presented as both an attacking and a defending position.

As was the case for the training positions, there are many ways to construct the testing positions, but each must have both an attacking and a defending solution. The testing positions included 20 defending positions and 20 attacking. In order for all of the positions to be presented in both attacking and defending forms, two testing sets were formed. These two sets were identical, including the order of positions, except that every position that appeared as attacking in one set appeared as a defending position in the other set.

The order in which positions appeared was determined once. Order was determined by random selection, but simpler transformations had to appear before a more complex transformation of the same training position. Attacking and defending positions alternated, with odd numbered trials being attacking in one testing set and even numbered trials being attacking in the other. An equal number of pin and fork moves had to be attacking and defending.

Because the two training positions the subject received were maximally dissimilar, it was clear which transfer condition a testing position belongs to for each training condition. There was always one training position that was only one factor (attack/defence, fork/pin, or rook/bishop) different from the manoeuver used in the testing position, while the other was different by two factors (except for the similar condition where one training position has all three factors the same). Hence one training position was always clearly more similar to a given testing position than was the other training position.

It was this more similar training position that determined to which transfer condition a testing position belonged.

The final position was constructed so as to be a plausible chess position but with no worthwhile captures possible for either side. However, a fatal knight move was available to one side that would fork the other side's king and queen. Such a move is extremely dangerous because it guarantees the capture of the queen, which is the most powerful piece on the board. There was also a distractor move available, involving an attack on the other queen that is pointless at best, but is actually a losing move.

In the attacking version the subject had the opportunity to make the fatal move. In the defending version the identical position was used but with the colors reversed so that now white (the subject) must try to avoid this dangerous move, a task that could be easily accomplished if the subject realizes that it must be done. The attacking version of the position is presented in figure 4 with white about to move.

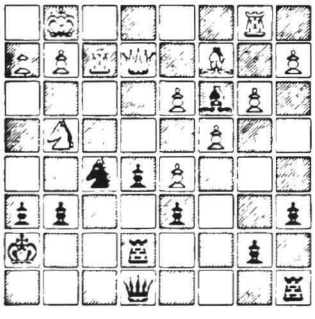


Figure 4. Final position (attacking version) with white about to move.

Procedure

Subjects were shown how the computer represented chess pieces and how to use the mouse to move pieces. They were told to follow the instructions that appeared on the screen, but that they could ask the experimenter questions if they were unclear with regard to any of the instructions.

The instructions first made clear that subjects would not see normal chess positions; in particular, kings were not special as they are in normal chess. Subjects were then given practice at detecting captures and were familiarized with using the mouse to move the pieces. Next, the subjects were presented with their attacking training position. They were to place the white piece on the board so as to guarantee the capture of a black piece no matter what move black subsequently made. It was emphasized that they could not place their piece on a square where it could immediately be captured. It was also made clear that it did not matter what happened to the white piece once it had captured a black piece, and therefore it was irrelevant if the piece they were to capture was protected by the other black piece.

The computer indicated if the subject's response was correct or incorrect. If an incorrect move was made, the computer displayed a correct response. This display lasted 10 s and showed the problem position with the white (attacking) piece placed on a correct square. To complete this stage subjects had to solve the position three times in a row. There were three different versions of the training position, each only differing from the others by a simple translation across the board.

The second set of practice trials presented subjects with their defending training position. Subjects were instructed that they were now to legally move one of the white pieces on the board so as to avoid the capture of a white piece. They were instructed that black was going to try to attack them in a similar way to that which the subject had used during the attacking training, but that they had a chance to pre-empt the placing of the attacking piece. If they made the right move at this stage then it would not matter where black placed their piece: black would not be able to force the capture of a white piece.

The link between attack and defence was made very explicit in the subjects instructions. They were advised to first think of where black was going to place their piece, and to then think how they could render that move harmless. It was also made clear that they had to be careful because even if they avoided one threat they might leave themselves open to another. Subjects were given similar feedback and the same completion criterion as they had for attacking training.

Subjects were then given the 40 testing trials from the randomly assigned testing set. No feedback was given at this stage, but subjects should have been able to recognize when they had made a correct move. After completing all testing trials subjects were told that they would be presented with a normal chess position and that they were to make the best move they could find. They were instructed that the relative value of pieces should be taken into account just as in normal chess (a factor that had been irrelevant up to this point). To guide them, a table giving the relative value of each piece

was displayed. Whether the subject saw the attacking or the defending position was crossed with assignment to the winning or the losing condition. Subjects in the winning condition were instructed, "In your position, white is WINNING and is on the ATTACK". Subjects in the losing condition were instructed, "In your position, white is LOSING and is on the DEFENSIVE".

Results

The defending positions in which a queen was the attacking piece were inherently very difficult, as reflected in very low solution rates and very long response times. For this reason these moves were eliminated from the analysis of response time data, as were the attacking versions of these positions.

The other positions had a very high solution rate, particularity the attacking positions with 98% correct solutions. Defending positions were obviously harder but still had an 82% solution rate. As error rates were low, response times were analyzed. Analyses of response time data did not distinguish between correct and incorrect solutions. In an essentially self-terminating task such as this, response times will be greater for incorrect solutions than for correct ones. If response times for incorrect moves were eliminated, information about difficulty would be lost (as long as subjects were motivated).

Table 1
Means and standard deviations (in parenthesis) of response times for each transfer condition and for attacking and defending positions in seconds, collapsed across all transformation positions.

	similar	attack/ defence	fork/ pin	rook/ bishop	Total
Attacking	15.6 (10.5)	17.1 (10.0)	21.2 (23.5)	16.6 (13.9)	17.6
Defending	29.8 (30.5)	36.4 (36.8)	31.1 (29.3)	37.4 (39.0)	33.7
Overall	22.7 (23.9)	26.8 (28.6)	26.2 (27.0)	27.0 (31.0)	25.7

The mean response times for the attacking and defending positions in each of the four transfer conditions are presented in Table 1, collapsed over all transformations. A 2x4 between-subjects analysis of variance revealed a significant main effect of whether the position was attacking or defending, $F(1,2440) = 225$, $p<.01$, ($MS_e = 703$). This simply confirms the expectation that the defending positions are more difficult than the attacking positions. The predicted main effect of transfer condition was obtained, $F(3,2440) = 3.48$, $p<.05$, as was a significant interaction between the two variables, $F(3,2440) = 5.45$, $p<.01$.

The nature of the interaction was examined by looking at the attacking and defending positions separately (see Table 1 for means). For attacking positions there was a significant effect of transfer condition ($F[3,1220] = 7.88$, $p<.01$, $MS_e = 238$). However, planned comparisons revealed only that the fork/pin condition was significantly slower than the similar condition ($F[1,1220] = 20.34$, $p<.01$). There were no significant differences between the similar condition and the attack/defence condition ($F[1,1220] = 1.39$, $p<.25$) or the bishop/rook condition ($F[1,1220] = .67$, *n.s.*).

For defending positions there was also an effect of transfer condition ($F[3,1220] = 3.77$, $p<.025$, $MS_e = 1167$). The planned comparisons revealed that the similar condition was significantly faster than the bishop/rook condition ($F[1,1220] = 7.55$, $p<.01$), and the attack/defence ($F[1,1220] = 5.78$, $p<.01$), but not significantly faster than the pin/fork condition ($F[1,1220] = .21$, *n.s.*).

In order to investigate the differences for particular transformation positions, pin positions were examined. Because every subject received every pin position (which was not the case for forks) a more powerful repeated measures analysis could be used. The mean response times for each pin transformation position for each transfer condition are presented in Table 2.

Table 2
Mean response times (s) for pin positions for each transformation position in each transfer condition.

	Transformation			
	identical	rotation	change-piece	change-structure
similar	15.3	19.3	26.4	27.4
defence/attack	19.7	28.3	30.2	29.5
pin/fork	22.4	30.0	31.4	28.2
bishop/rook	21.2	29.2	28.1	23.3

For the identical transformation a multivariate analysis of variance revealed a significant effect of transfer, $F(3,69)=5.88$, $p<.01$. Planned univariate contrasts revealed significant differences between the similar condition and each of the other three transfer conditions: for similar with defence/attack, $F(1,71)=4.54$, $MS_e=160$, $p<.05$; with pin/fork, $F(1,71)=8.19$, $MS_e=226$, $p<.01$; and with bishop/rook, $F(1,71)=8.23$, $MS_e=155$, $p<.01$. Similar results were obtained for the rotation transformation in which a significant effect of transfer condition was obtained, $F(3,69)=5.49$, $p<.01$.

Planned univariate contrasts showed significant differences between the similar condition and each of the other three transfer conditions: for similar with defence/attack, $F(1,71)=4.65$, $MS_e=627$, $p<.05$; with pin/fork, $F(1,71)=6.51$, $MS_e=636$, $p<.05$; and with bishop/rook, $F(1,71)=7.84$, $MS_e=452$, $p<.01$. However there were no effects of transfer condition for the change-piece ($F[3,69]=.26$, *n.s.*) or the change-structure ($F[3,69]=.77$, *n.s.*) transformations.

Responses on the final position were analyzed by classifying all moves as correct or incorrect. A correct move in the attacking condition (when the subject had a chance to make the fatal move) was only recorded if they made the knight forking move. A correct move in the defending position (when the subject is threatened by the fatal move) is any move that renders the knight move harmless.

Table 3
Number of correct and incorrect solutions given to the final position in the attacking condition as a function of subject being given winning or losing manipulation.

	attacking position		defending position	
	winning	losing	winning	losing
Correct	5	9	0	4
Incorrect	6	8	16	9

The frequencies for correct and incorrect moves, for both attacking and defending positions, as a function of whether the subject was told that they were winning or losing, are presented in Table 3. There were no significant differences for attacking positions, $X^2(1) = .15$, *n.s.*, however the defending position yield a significant difference ($X^2[1] = 9.16$, $p<.005$) indicating that subjects were more likely to solve the problem, by recognizing the need for defence, when they were told that they were losing, rather than winning.

Discussion

These results provide support for the claim that goal similarity plays a role in analogical problem solving. In doing so they indicate that this task may be a useful one for further investigating the role of goal similarity in analogical processes. The overall findings, and those from the similar and rotation transformation positions, indicated that response times were faster if goals were kept consistent between the training and testing positions. This result is strengthened by the fact that the similar position appeared after many other attacking and defending positions had been seen and solved by the subjects. The fact that the conditions that we would expect to be hard to map, the fork/pin and bishop/rook conditions, were also generally slower then the similar condition supports the contention that mapping is being used to help solve positions.

The lack of a transfer position effect for the change-piece transformation positions is interesting. It appears to indicate that changing a piece is more than a simple semantic change. It may constitute a structural change, due to the fact that different pieces move differently. Hence changing a piece may change how the pieces relate to each other, even without changing their positions.

It is still not conclusively established, however that ease of analogical retrieval or mapping caused the faster responses in the similar conditions. Further research should try to establish directly how well subjects retrieve or map to these positions. Including a recognition task would be one way to do this. Another way to make the task more sensitive to mapping effects is to present ambiguous positions with multiple valid solutions and observe if subjects favor solutions that map most closely to their training positions

The finding that the goals at the time of test can influence solutions provides a powerful demonstration of the effects of influencing the subjects goals. This is particularly so since no other changes are made and the information that is provided is not directly relevant to determining the optimal move. However it is possible that this result could be explained in terms of demand characteristics. In future research this problem could be addressed by using less explicit instructions. It should also be noted that in this experiment the final position was not designed as an analogical task, as there is nothing provided as an explicit source analogue. Future studies could turn this into an analogical task by attempting to provide an explicit source. How well subjects could find such embedded analogues is unclear, but this ability could be examined by using a recognition task in which subjects would have to recognize an embedded analogy present in a full position (again goal similarity, structural, and semantic factors could be again manipulated).

This study did not examine skilled chess play; however the results have implications for chess skill, if they prove to be extendable to skilled chess players. If chess skill is based on analogical processes, then it would provide an explanation for the characteristic of chess masters that deGroot (1965) found that most distinguished them from less skilled players: that they simply selected better moves to examine in the first place. Because neither he nor the players were able to explain this,

deGroot ascribed it to intuition, but its almost perceptual quality seems to fit quite well into an analogical framework.

In conclusion, this study provided support for the claim that goal similarity plays a role in problem solving, in that responses were made more quickly if consistent goals were given. This advantage may well be due to the improvement in the ease of analogical mapping that consistent goals provided, although further research is required to establish this with more confidence. The fact that this task is sensitive to such effects may mean it can be used to test different models of analogical mapping and problem solving. Such models often differ crucially in the ways they handle goals.

References

Brown, A. L., Kane, M. J., & Echols, C. H. (1986). Young children's mental models determine analogical transfer across problems with a common goal structure. *Cognitive Development, 1*, 103-121.

Carbonell, J. (1983) Learning by analogy: Formulating and generalizing plans from past experience. In R. Michalski, J. Carbonell, & T. Mitchell, (Eds.), *Machine learning: An artificial intelligence approach*. Los Altos: Tioga, 137-161.

Carbonell, J. (1986) Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In R. Michalski, J. Carbonell, & T. Mitchell (Eds.), *Machine learning: An artificial intelligence approach*, vol 2. Los Altos: Morgan Kaufman, 371-392.

deGroot, A. D. (1965). *Thought and choice in chess*. The Hague: Mouton.

Gentner, D. (1989). The mechanisms of analogical learning. In S. Vosniadou & A. Ortony (Eds.), *Analogy, similarity, and thought*. Cambridge, Mass.: Cambridge University Press.

Gentner, D., & Toupin, C. (1986). Systematicity and surface similarity in the development of analogy. *Cognitive Science, 10*, 277-300.

Gick, M. L., & Holyoak, K. J. (1980). Analogical problem solving. *Cognitive Psychology, 12*, 306-355.

Gick, M. L., & Holyoak, K. J. (1983). Schema induction and analogical transfer. *Cognitive Psychology, 15*, 1-38.

Hammond, K. (1989). *Case-based planning*. Boston: Academic Press.

Holyoak, K. J., & Koh, K. (1987). Surface and structural similarity in analogical transfer. *Memory & Cognition, 15*, 332-340.

Holyoak, K. J., & Thagard, P. (1989). Analogical mapping by constraint satisfaction. *Cognitive Science*, 13, 295-355.

Schank, R. C. (1982). *Dynamic memory*. Cambridge: Cambridge University Press.

Ross, B. (1987). This is like that: The use of earlier problems and the separation of similarity effects. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 13*, 629-639.

Ross, B. (1989). A further analysis of the access and use of earlier problems: Distinguishing different superficial similarity effects. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15*, 456-468.

Winston, P. H. (1980). Learning and reasoning by analogy. *Communications of the ACM, 23*, 689-703.

Winston, P. H. (1982). Learning new principles from precedents and exercises. *Artificial Intelligence, 19*, 321-350.

Internal Analogy: A Model of Transfer within Problems

Angela Kennedy Hickman
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
ach@ml.ri.cmu.edu

Jill H. Larkin
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213
larkin@tulip.psy.cmu.edu

Abstract

Understanding problem solving and methods for learning is a main goal of cognitive science. Analogical reasoning simplifies problem solving by transferring previously learned knowledge from a source problem to the current target problem in order to reduce search. To provide a more detailed analysis of the mechanisms of transfer, we describe a process called *internal analogy* that transfers experience from a completed subgoal in the same problem to solve the current target subgoal. We explain what constitutes an appropriate source problem and what knowledge to transfer from that source, in addition to examining the associated memory organization. Unlike case-based reasoning methods, this process does not require large amounts of accumulated experience before it is effective; it provides useful search control at the outset of problem solving. Data from a study of subjects solving DC-circuit problems designed to facilitate transfer supports the psychological validity of the mechanism.

1. Introduction

Analogical reasoning is an effective method of recycling past experience to guide problem solving. To begin the analysis of this process, we formulate the following five steps. First, the problem solver must determine a set of candidate sources. The utility of the procedure relies on the identification of the relevant knowledge. Second, one solution must be retrieved to function as the actual source. Third, the source solution must be reinstantiated and modified to solve the target problem. Fourth, the new solution should be stored so that the problem solver can reason from it to solve future problems. Fifth, the problem solver should receive some knowledge of results concerning the effort required to perform the analogy and the successfulness of the procedure. This information can be used to provide feedback to the retrieval steps of the process.

Although analogy has been explored previously [1, 2, 7, 8, 16, 17], most of the work has focused on the mapping procedure outlined in step three above. [11], [12] and [19] have addressed memory organization (related to steps one, two, and four above) in detail, but their ideas have not been integrated with an analogical mechanism. ICARUS [14] and EUREKA [10] have incorporated the whole process to some extent. However, neither system embeds its solution within a general implementation of a problem solver (for example, EUREKA is not capable of backtracking). As a result, they cannot solve problems which are as difficult as those reported in this study.

This paper addresses the first four steps of the process with a transfer mechanism called *internal* analogy, which works on similar subgoals of a single problem. This is in contrast to *within-domain* and *cross-domain* analogy which transfer knowledge across separate problems from the same domain and different domains, respectively. Work in progress using derivational analogy [4] in PRODIGY [3, 22] is beginning to address most of the steps above, but for cross-problem, within-domain analogy.

Our internal analogy mechanism is tightly integrated into a general problem solver for the physical sciences, RFERMI, and is effective in reducing search [9]. Unlike case-based reasoning methods, the process does not require large amounts of accumulated experience before it is effective; it provides useful search control at the outset of problem solving. In addition, psychological predictions drawn from the computational model of internal analogy were supported by data from a study of subjects solving DC-circuit and fluid statics problems that were designed to facilitate transfer.

The next section presents the implementation of the internal analogy algorithm in RFERMI, as well as an example trace of the non-learning system. Section 3 contains the psychological predictions derived from the computational model and the analysis of the data we collected. We conclude with a discussion evaluating internal analogy.

2. Computational Model

The internal analogy process described in the preceding section is implemented in a problem solver for the physical sciences named RFERMI. This system is a rule-based version of the FERMI system [15] and is based in part on studies of effective representations and methods for solving physics problems [5, 18]. Its task

domains range from linear algebra and DC-circuits to fluid statics and classical mechanics. RFERMI maintains a principled decomposition of knowledge in order to retain the power of its domain specific knowledge while utilizing the cross-domain applicability of its more general knowledge.

RFERMI's declarative knowledge of scientific principles is organized in a *quantity hierarchy* which is stored as frames. The frame system used to implement the hierarchy is a component of FRulekit [20], a forward chaining production system. Through the use of inheritance, this hierarchy efficiently stores knowledge about quantities such as resistances, pressure drops, potential drops, and two dimensional areas.

RFERMI's procedural knowledge, as organized in its *actions hierarchy*, is of two types. First, domain specific knowledge is stored in *puller* frames that are interpreted into FRulekit rules. These pullers encode equational knowledge, such as Ohm's law, and procedures, such as those for finding the pressure drop between two points in a static liquid or the electro-magnetic force of a battery. Second, RFERMI's more general and widely applicable knowledge, such as its iterative decomposition procedure, constitute its *methods*. Methods are associated with generalized quantities so that a quantity inherits access to a method from superordinate quantities. For example, potential drop inherits access to the path invariance method from scalar field difference, which is a generalized quantity. Therefore, the method for equating potential drops along two alternate paths can be used to solve for a potential drop.

Although the system is implemented in a forward chaining production system, it maintains a backward chaining control structure via the goal monitor. The space it searches while solving for an unknown quantity is structured in a traditional AND-OR manner. When there are multiple means for pursuing a goal (i.e., pullers and methods) an OR node is generated, and when a method spawns a conjunctive set of unmet subgoals (i.e., unknown quantities) an AND node is generated. RFERMI organizes its search in a manner combining depth-first and breadth-first expansion. At OR nodes, it contains heuristic preferences for using the specific puller knowledge over the more general methods.

As an example, we present RFERMI's problem solving without learning on the problem in Figure 2-1(see Figure 2-2). The system first solves for I_2 by applying a puller. Lines 7-14 show that it solves for R_1 by generating subgoals for V_{ab} and I_1. V_{ab} is solved using a puller, which results in R_1's solution. In line 16, RFERMI chooses a different puller to solve I_3 than it used to solve I_1. (It could reduce the required search by solving I_3 in a similar manner since the goals are analogical and the system solved I_1 with relatively little effort.) In both cases, both pullers were applicable, and the system chose randomly. However, this line of reasoning ends in goal circularity (see line 20) since V_{cd} has already been posted. In lines 21-23, it tries another instantiation of the same puller, which also results in goal circularity. Finally, in lines 24-29, it obtains the value of V_{cd} by using the puller that solved I_1. The end of the trace shows how R_4 is solved.

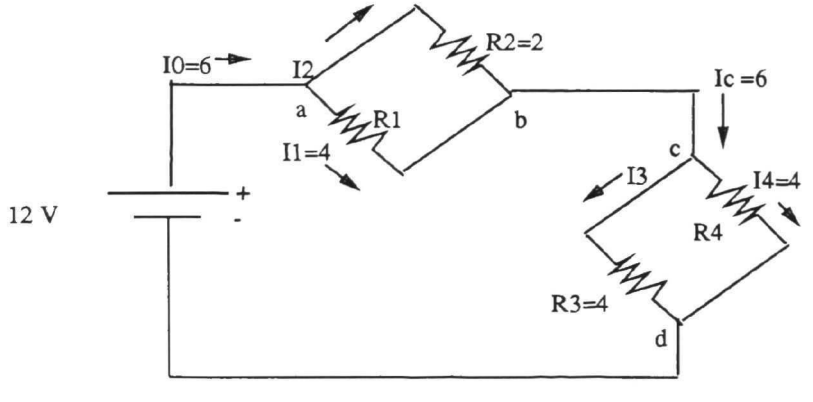


Figure 2-1: Problem 5.

The implementation of internal analogy in RFERMI relies on the trace facility of the problem solver. While solving a problem, the system stores the expanded goal tree along with the status of each goal (succeeded, failed, or pending). For goals that succeed, the instantiated productions that achieved the goal are also stored. As shown by the internal analogy algorithm in Figure 2-3, when the system begins to solve a new target subproblem, it checks if it has previously solved a goal of the same quantity type. (For example, I_1 is of the same quantity type as I_3 in Figure 2-1 since they are both currents.) If so, the most recently solved

1. Post Goals:	I_2, R_1, I_3, R_4	
2. Choose Goal:	I_2	
3. Apply Puller:	$I_2 = I_0 - I_1$	current flowing into a = current flowing out of a
4. Lookup:	$I_0 = 6$	
5. Lookup:	$I_1 = 4$	
6. Compute:	$I_2 = 2$	
7. Choose Goal:	R_1	
8. Apply Puller:	$R_1 = V_{ab} / I_1$	Ohm's law (V=IR)
9. Apply Puller:	$V_{ab} = I_2 R_2$	Ohm's law (V = IR)
10. Lookup:	$I_2 = 2$	
11. Lookup:	$R_2 = 2$	
12: Compute:	$V_{ab} = 4$	
13. Lookup:	$I_1 = 4$	
14. Compute:	$R_1 = 1$	
15. Choose Goal:	I_3— internal analogy would apply	
16. Apply Puller:	$I_3 = V_{cd} / R_3$	Ohm's law (V = IR)
17. Apply Puller:	$V_{cd} = I_4 R_4$	Ohm's law (V = IR)
18. Lookup:	$I_4 = 4$	
19. Apply Puller:	$R_4 = V_{cd} / I_4$	Ohm's law (V = IR)
20. Fail Circular Goal:	V_{cd}	
21. Apply Puller:	$V_{cd} = I_3 R_3$	Ohm's law (V = IR)
22. Apply Puller:	$I_3 = V_{cd} / R_3$	Ohm's law (V = IR)
23. Fail Circular Goal:	V_{cd}	
24. Apply Puller:	$I_3 = I_c - I_4$	current flowing into c = current flowing out of c
25. Lookup:	$I_c = 6$	
26. Lookup:	$I_4 = 4$	
27. Compute:	$I_3 = 2$	
28. Lookup:	$R_3 = 4$	
29. Compute:	$V_{cd} = 8$	
30. Lookup:	$R_3 = 4$	
31. Compute:	$I_3 = 2$	
32. Choose Goal:	R_4— internal analogy would apply	
33. Apply Puller:	$R_4 = V_{cd} / I_4$	Ohm's law (V = IR)
34. Lookup:	$V_{cd} = 8$	
35. Lookup:	$I_4 = 4$	
36. Compute:	$R_4 = 2$	

Figure 2-2: RFERMI's behavior on Problem 5 with no learning.

instance is chosen as the *candidate* source. If the candidate source succeeded and no more information was known about it than is known about the current problem, then it is chosen as the *actual* source. Otherwise, it is rejected because the system had additional knowledge during the previous problem solving which may have been crucial to its success. Without that knowledge, it may be unable to recycle the old solution to solve the current target problem. The failure case is decided in just the opposite manner. If the candidate source failed and contained no less information than the current problem, it is chosen. Otherwise, it is

rejected. This check ensures that RFERMI does not choose a source that failed because there was less information available to the problem solver and prematurely fail the current target as a result. The mapping proceeds based on the success or failure of the source. If the source failed, then the processing of the current goal is suspended and another goal is explored. If all other problem solving fails, this goal may be later reopened. If, on the other hand, the source succeeded, its solution is appropriately reinstantiated for the current subgoal, and the solution is replayed. Since this newly solved subgoal is contained in the current problem, it is available to the algorithm as a future candidate source.

```
1. IF there are untested previously explored subgoals of the same type as target
2.         THEN candidate := most recently explored same-type subgoal
3.         ELSE fail internal-analogy
4. IF candidate succeeded
5.         THEN IF information-content(candidate) <= information-content(target)
6.                   THEN source := candidate & reinstantiate the source solution
                       to solve the target
7.                   ELSE tested(source) := TRUE & internal-analogy(target)
8.         ELSE IF information-content(candidate) >= information-content(target)
9.                   THEN source := candidate & suspend(target)
10.                  ELSE tested(source) := TRUE & internal-analogy(target)
```

Figure 2-3: The internal analogy algorithm.

Steps 5 and 8 of the algorithm in Figure 2-3 compare the amount of information known about a candidate goal at the time it was solved with the amount of information known about the current target goal. This comparison is carried out in RFERMI by calculating the set of variables in the left hand sides of the rules that solved the candidate source goal and that had known values. We call this the *information content* of the candidate source. The information content of the current target goal is computed by calculating the set of these same variables that have known values in the current working memory. For example, suppose the candidate source goal is to find the potential drop between points *a* and *b* in a circuit, and the resistance and the current between those two points were known. Now suppose the current target goal is to find the potential drop between points *c* and *d* in the same circuit, and the equation that solved the candidate source goal was potential-drop = current * resistance. If the resistance between *c* and *d* is known but the current is not, then the information content of the candidate source goal is said to be greater than that of the target goal (because the current was known in the candidate source goal).

Step 6 of Figure 2-3 reinstantiates and replays the solution of the source problem in order to solve the target problem. RFERMI carries out this step by instantiating the productions that solved the source goal in the current working memory and applying them. We call this *operator-driven mapping*, and it answers the important question that Structure Mapping [7] inadvertently poses: how does one identify the salient structure to map? We operationally define the salient structure to be the relevant variables that are tested in the left-hand sides of the operators that solved the source goal.

As a last comment, we mention that it is especially important to only suspend the processing in Step 9 of Figure 2-3. The system cannot terminally fail the goal because one of RFERMI's other general methods may still solve the problem, although more expensively.

The internal analogy mechanism described above has proven to be an effective learning mechanism in RFERMI. Detailed theoretical and empirical analyses of the search reduction it provides are described in [9].

3. Protocol Data

If, as hypothesized, the internal analogy mechanism embedded in RFERMI has any psychologically validity, then the computational model described in the previous section predicts that subjects will exhibit the following behaviors during problem solving:

- Knowledge will be transferred from either previously successful or previously failed goals.
- The source will be of the same quantity type as the target and have a compatible information content.
- Problem solving using analogy will require less effort (search) than would otherwise be necessary.
- Transfer from previously failed problem solving will enable the subject to know that a particular procedure as instantiated at the current point will fail. Thus, he should choose a different procedure.

- Transfer from previously successful problem solving will allow the subject to know precisely which procedures to choose to calculate the quantity and all its subquantities. For problems in the physical sciences, this means that the subject should know which equations to reuse to solve the unknown. However, these equations must be reinstantiated to reflect the new problem solving context.
- Since RFERMI randomly selects among its applicable pullers for solving any given subgoal, we predict that subjects will show individual differences in their problem solving behavior. The system also has two different control strategies: one strictly depth first and the other more breadth first. It's problem solving differs according to the current control strategy. As a result, we predict further individual differences will arise from the subjects' varied control strategies.

To test the predictions, we studied subjects solving problems from two of RFERMI's task domains, DC-circuits and fluid statics. These problems were designed to facilitate three kinds of transfer: internal analogy, within-domain analogy, and cross-domain analogy. The four subjects had all earned an A or a B in a year-long college physics course, but they had not solved any problems in these domains for several years. We chose subjects with this level of proficiency because we believed that they would be the most likely to exhibit the desired transfer. Subjects with a high level of expertise tend to use compiled knowledge rather than analogical reasoning; subjects with very little expertise tend to use brute force search. The subjects were given a remedial, which was in a two-column format, covering the knowledge necessary for the experiment. The left column contained the circuit information, and the right column contained the fluid statics information. Analogical concepts were presented directly across the page from each other. In order to verify the remediation, the subjects were asked to explain sparse written solutions to three example problems. These example problems were also designed to serve as analogical sources for the five problems that the subjects were asked to solve next.

We observed all three types of transfer. However, the more local types of transfer happened more frequently; only one instance of cross-domain transfer occurred. Due to space limitations, we discuss the subjects' behavior only on Problem 5, which was shown in Figure 2-1 and designed to facilitate internal analogy. Below we demonstrate that RFERMI with the internal analogy mechanism models the subjects' behavior well. The mechanism also reduces the search previously required to solve I_3 and R_4 by about 50% (compare lines 15-36 of Figure 2-2 with lines 16-30 of Figure 3-1).

We begin by comparing the behavior produced by RFERMI with internal analogy to that of Subject 1 on the example problem (see Figure 3-1). Problem solving for both proceeds similarly, except for the following differences which are unimportant with respect to the analogical mechanism. Between lines 2 and 3 of the figure, Subject 1 does some erroneous problem solving and decides to start over. As can be seen in lines 3 and 18 of the protocol, the system always posts an equation with the desired unknown on the left hand side, while the subject posts the version of the current invariance equation that corresponds to the associated prose in the remedial. In lines 8 and 10 of the protocol, Subject 1 posts incorrect equations; this will have interesting side effects later in the problem solving. The problem solver represents potential drops as a drop between two points, regardless of the path. The subject, however, clearly distinguishes potential drops with the same endpoints over different paths; this leads to his extra step in line 9. Occasionally, Subject 1 will take an arithmetical "shortcut" by not restating the implied left-hand side of the equation or by reducing fractions to their lowest terms (lines 11-12, 26-27 and 29).

Ignoring these small differences, RFERMI models the subject extremely well. Both solve equations for I_2 and R_1 in a straightforward manner. At line 17, the system's analogical mechanism is invoked because it is has solved a goal of the same quantity type with a compatible information content, I_2 (I_0 and I_1 are known while I_2 is unknown at line 3, and I_c and I_4 are known while I_3 is unknown at line 18). It retrieves the productions that solved I_2 and reinstantiates them. This saves the system search time in two ways. First, it does not have to compute which productions to apply— the analogy mechanism specifies them. Second, there are other applicable pullers at this point that would require more problem solving effort if they were used, as shown in lines 15-31 of Figure 2-2. Subject 1 also recognizes that I_3 is an analogical goal to I_2 at line 17: he states, "this (pointing to I_3) is just like that (pointing to I_2)". Then, he quickly reinstantiates the equation that he used in line 3 and solves for I_3. Similar recycling of past experience occurs for both the problem solver and the subject in lines 22-30 while solving for R_4. When Subject 1 says, "back to here," in these lines, he is pointing to the equation $R_1 = I_1/V_{r1}$ which he wrote in line 8. The subject analogizes from the incorrect equations in lines 8 and 10 and reuses them in lines 24 and 25, which causes him to derive an incorrect answer for R_4. Had he used the correct equations earlier, he would have solved this problem

	FERMI	Subject 1	
1. Post Goals:	**I_2, R_1, I_3, R_4**	**I_2, R_1, I_3, R_4**	
2. Choose Goal:	**I_2**	**I_2**	
3. Apply Puller:	$I_2 = I_0 - I_1$	$I_0 = I_1 + I_2$	
4. Lookup:	$I_0 = 6$	$I_0 = 6$	
5. Lookup:	$I_1 = 4$	$I_1 = 4$	
6. Compute:	$I_2 = 2$	$I_2 = 2$	
7. Choose Goal:	**R_1**	**R_1**	
8. Apply Puller:	$R_1 = V_{ab} / I_1$	$R_1 = I_1 / V_{r1}$	
9. Apply Method:		$V_{r1} = V_{r2}$[1]	
10. Apply Puller:	$V_{ab} = I_2 R_2$	$V_{r2} = I_2 / R_2$	
11. Lookup:	$I_2 = 2$	$= 2/$	
12. Lookup:	$R_2 = 2$	2	
13. Compute:	$V_{ab} = 4$	$V_{r1} = 1$	
14. Lookup:	$I_1 = 4$	$I_1 = 4$	
15. Compute:	$R_1 = 1$	$R_1 = 4$	
16. Choose Goal:	**I_3**	**I_3**	
17. Analogize:	*Fires analogy to I_2*	*"This is just like that."*	
18. Apply Puller:	$I_3 = I_c - I_4$	$I_c = I_3 + I_4$	
19. Lookup:	$I_0 = 6$	$I_c = 6$	
20. Lookup:	$I_4 = 4$	$I_4 = 4$	
21. Compute:	$I_3 = 2$	$I_3 = 2$	
22. Choose Goal:	**R_4**	**R_4**	
23. Analogize:	*Fires analogy to R_1*	*"Back to here."*	
24. Apply Puller:	$R_4 = V_{cd} / I_4$	$R_r = I_4 / V_{r4}$	
25. Apply Puller:	$V_{cd} = I_3 R_3$	$V_{r3} = I_3 / R_3$	
26. Lookup:	$I_3 = 2$	$= 1/$	
27. Lookup:	$R_3 = 4$	2	
28. Compute:	$V_{cd} = 8$	$V_{r4} = 1/2$	:Apply Method
29. Lookup:	$I_4 = 4$	$4/.5 = 8$	:Compute
30. Compute:	$R_4 = 2$	$R_4 = 8$	

Figure 3-1: The behavior of RFERMI with learning and Subject 1 on Problem 5.

correctly.

All four subjects performed internal analogy on Problem 5, but each exhibited a different control structure. Subject 1 backward chained much like RFERMI, while Subject 4 demonstrated more expertise in the domain and forward chained. This behavior is consistent with the results reported in [21] that show that experts tend to forward chain in search spaces that they expect to be small. Subject 2 began by backward chaining and switched to forward chaining as he gathered more experience in the domain. Subject 3 struggled to complete the problem and explored the subgoals in a nonstandard order. We now examine each of the other subject's problem solving more closely.

In contrast to Subject 1, Subject 4 solves the problem by forward chaining. This subject maps the

[1]The potential drop between any two points is the same regardless of the path chosen between the points.

analogical subgoals explicitly by their quantity type and information content. At the outset of the problem solving, he says, "So, I have two resistors where the current is given and the resistance is left unknown (R_1 and R_4) and two resistors where the resistance is given and the current is left unknown (R_2 and R_3)". He proceeds to solve for I_2 and R_1. At this point he says, "similar situation here," and solves for I_3, reusing the the equation that solved I_2 reinstantiated for the current goal. In a similar fashion, he uses the equations that solved R_1 to solve R_4. This subject, like Subject 2 and Subject 1 on other problems, tends not to verbalize the uninstantiated equation during the replay of the analogy but verbalizes the instantiated form instead.

Subject 3, who begins backward chaining and switches to forward chaining, states early in his problem solving that $I_1=I_4=4$ and $I_0=I_c$. When he solves I_2, he immediately states the same answer for I_3, without additional computation. It appears that his analogical reasoning is more advanced than RFERMI's. In addition to reposting and reinstantiating equations, this subject is able to recognize when the relevant variables have exactly the same value, and the answer can be recycled directly. With this straightforward extension added to the system, it could gain an even greater reduction in search. The point at which Subject 3 switches to forward chaining is also significant: he finishes solving for V_{ab}, and he recycles the equation he used in a newly instantiated form for V_{cd}. The switch from backward chaining to forward chaining seems to be triggered, at least in part, by an analogical goal.

Subject 2 is considerably less skilled at solving these types of problems than the other subjects. He struggles to solve any of the subgoals using the same knowledge encoded in RFERMI's pullers, methods, and algebra module. When he does finally solve I_2 and R_1, however, he immediately restates the current invariance relation and quickly solves I_3 and R_4. There is nothing in his analogical transfer that we did not observe in the previous three protocols.

4. Discussion and Conclusions

Although the subjects showed individual differences in their control strategies, the basic components of the analogical reasoning were those that the computational mechanism predicted. The subjects transferred knowledge from successful problem solving in order to reduce the effort required to solve the target subgoal. They simply reposted the previously successful equations and reinstantiated them in the current context. In every instance, the sources and targets were of the same quantity type and had compatible information contents.

The system models Subject 1 well in its current state. With a forward chaining control strategy, it could easily model Subject 4 as well. To model Subject 2, one additional capability must be added to the system: it should recognize those occasions when the equations need not be reinstantiated but the value may be directly recycled. Even though the computational model focused our attention toward particular problem solving behavior in the protocols, the protocols continue to suggest useful extensions to the system.

Relaxing the notion of compatible information content will provide internal analogy with a more flexible matching mechanism than either SOAR's chunking [13] or macro-operators [6] possesses. This will allow our learning method to provide search control when the others cannot. Extending the implementation to effect within-domain and cross-domain transfer will also increase its utility.

In conclusion, our study of internal analogy has described a new process for transferring knowledge within a single problem. It has also provided a more complete analysis of the processes needed for analogical transfer than has previously been presented. We specify what constitutes an appropriate source and what knowledge to transfer to the target. In addition, the mechanism is tightly integrated into a general problem solver and does not require an alternate reasoning engine or large case libraries. The psychological validity of the mechanism has also been supported through the psychological data presented.

Acknowledgements This research was supported in part by DARPA contract number F33615-87-C-1499 Amendment 20, and AIP-ONR contract number N00014-86-K-0678-N123. Angela Kennedy Hickman was also partially supported during this research by a Zonta International Amelia Earhart fellowship. We would like to thank Peter Shell for his invaluable contribution to the implementation. Jaime Carbonell, Jill Fain Lehman and David Plaut increased the clarity and the content of the paper through their comments. Sandra Esch and David Plaut assisted in setting up the lab for the experiments.

References

1. Adelson, Beth, "Analogical Reasoning: Problem-Driven Mapping and Debugging," Tech. report, Department of Computer Science, Tufts University, 1989.

2. Anderson, J. R., "A Theory of the Origins of Human Knowledge," *Artificial Intelligence,* Vol. 40, 1989, pp. 313-351.
3. Carbonell, J. G. and Veloso, M. M., "Integrating Derivational Analogy into a General Problem Solving Architecture," *Proceedings of the First Workshop on Case-Based Reasoning,* Morgan Kaufmann, May 1988.
4. Carbonell, J. G., "Derivational Analogy: A Theory of Reconstructive Problem Solving and Expertise Acquisition," in *Machine Learning, An Artificial Intelligence Approach, Volume II,* Michalski, R. S., Carbonell, J. G. and Mitchell, T. M., eds., Morgan Kaufmann, 1986, pp. 371-392.
5. Chi, M. T. H., Feltovich, P. J. and Glaser, R., "Categorization and representation of physics problems by experts and novices," *Cognitive Science,* Vol. 5, 1981, pp. 121-152.
6. Fikes, R. E. and Nilsson, N. J., "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving," *Artificial Intelligence,* Vol. 2, 1971, pp. 189-208.
7. Gentner, D., "Structure mapping: A theoretical framework for analogy," *Cognitive Science,* Vol. 7, 1983, pp. 155-170.
8. Gick, M. L. and Holyoak, K. J., "Analogical Problem Solving," *Cognitive Psychology,* Vol. 12, 1980, pp. 306-355.
9. Hickman, A. Kennedy, Shell P. and Carbonell, J. G., "Internal Analogy: Reducing Search during Problem Solving," *Proceedings of AAAI-90,* 1990, (submitted).
10. Jones, R., *A Model of Retrieval in Problem Solving,* PhD dissertation, Department of Information and Computer Science, University of California, Irvine, 1989.
11. Kolodner, J. L., "Maintaining Organization in a Dynamic Long-Term Memory," *Cognitive Science,* Vol. 7, 1983, pp. 243-280.
12. Kolodner, J. L., "Reconstructive Memory: A Computer Model," *Cognitive Science,* Vol. 7, 1983, pp. 281-328.
13. Laird, J. E., Rosenbloom, P. S. and Newell, A., "Chunking in SOAR: The Anatomy of a General Learning Mechanism," *Machine Learning,* Vol. 1, No. 1, 1986, pp. 11-46.
14. Langley, P., Thompson, K., Iba, W. F., Gennari, J. and Allen, J. A., "An Integrated Cognitive Architecture for Autonomous Agents," Tech. report 89-28, Department of Information and Computer Science, University of California, Irvine, 1989.
15. Larkin, J. H., Reif, F., Carbonell, J. G. and Gugliotta, A., "FERMI: A Flexible Expert Reasoner with Multi-Domain Inferencing," *Cognitive Science,* Vol. 12, 1988, pp. 101-138.
16. Lewis, C., "Why and How to Learn Why: Analysis-based Generalization of Procedures," *Cognitive Science,* Vol. 12, 1988, pp. 211-256.
17. Pirolli, P., "A Model of Purpose-driven Analogy and Skill Acquisition in Programming," *The 9th Annual Conference of The Cognitive Science Society,* Lawrence Earlbaum Associates, 1987, pp. 609-621.
18. Reif, F. and Heller, J. I., "Knowledge structure and problem solving in physics," *Educational Psychologist,* Vol. 17, 1982, pp. 102-127.
19. Schank, R. C., "Language and Memory," *Cognitive Science,* Vol. 4, 1980, pp. 243-284.
20. Shell, P. and Carbonell, J. G., "The FRuleKit Reference Manual", CMU Computer Science Department internal paper.
21. Simon, D. P. and Simon, H. A., *Individual Differences in Solving Physics Problems,* Lawrence Earlbaum Associates, Hillsdale, New Jersey, 1978, ch. 13.
22. Veloso, M. M. and Carbonell, J. G., "Learning Analogies by Analogy - The Closed Loop of Memory Organization and Problem Solving," *Proceedings of the Second Workshop on Case-Based Reasoning,* Morgan Kaufmann, May 1989.

Making SME greedy and pragmatic

Kenneth D. Forbus Dan Oblinger
Qualitative Reasoning Group

Beckman Institute, University of Illinois
Phone: (217) 333-0193; Internet: forbus@p.cs.uiuc.edu

Abstract: The Structure-Mapping Engine (SME) has successfully modeled several aspects of human analogical processing. However, it has two significant drawbacks: (1) SME constructs all structurally consistent interpretations of an analogy. While useful for theoretical explorations, this aspect of the algorithm is both psychologically implausible and computationally inefficient. (2) SME contains no mechanism for focusing on interpretations relevant to an analogizer's goals. This paper describes modifications to SME which overcome these flaws. We describe a *greedy merge algorithm* which efficiently computes an approximate "best" interpretation, and can generate alternate interpretations when necessary. We describe *pragmatic marking*, a technique which focuses the mapping to produce relevant, yet novel, inferences. We illustrate these techniques via example and evaluate their performance using empirical data and theoretical analysis.

1 Introduction

The importance of analogy in human reasoning makes it a natural focus for cognitive simulation. The Structure-Mapping Engine (SME)[6,7], has been used to successfully model several aspects of human analogical processing. As a simulation of Gentner's Structure-Mapping theory [9,10], SME has been used to model human soundness judgements [13], to study the representational and processing choices in analogical processing [8], and as part of a model of sequence learning [14]. SME has also been used in an AI system which learns qualitative physics by analogy [3].

We believe several features of SME are accurate reflections of human analogical processing, including the emergence of global interpretations from local matches, the use of structural evaluation criteria as a default means of judging a comparison, the ability to generate novel candidate inferences, and the ability to construct and compare multiple interpretations of a comparison. However, the current SME algorithm has several drawbacks. First, SME constructs all structurally consistent interpretations of an analogy. This is often useful for theoretical explorations, since it allows one to know for certain the best possible interpretation of a given comparison. But it is extremely implausible psychologically. There are in the worst case a factorial number of potential solutions, making exhaustive enumeration impossible under any reasonable assumptions about human processing constraints. Even for theoretical explorations, as we tackle more realistic representations (c.f., [8]) this aspect of SME has become a stumbling block. To use SME as a central component in larger-scale simulations and AI systems, a more practical algorithm is needed. Here we describe a *greedy algorithm* which efficiently provides good approximations to the "best" interpretation. Although any greedy algorithm must sometimes fail to deliver optimal solutions, we demonstrate that in fact on this task it performs superbly.

The second drawback is that SME does not focus the mapping process according to the goals of the system. Such influences can be incorporated into analogical processing in several ways. Standard structure-mapping postulates that goals help determine both what gets matched and how the match is evaluated, but excludes them from the mapping stage itself [11]. Holyoak and Thagard's ACME model [12] blends structural, semantic, and pragmatic considerations into weights in a connectionist network, using a relaxation scheme to derive a single solution as an approximate best mapping. In addition to biasing preference for correspondences according to relevance, they allow *queries* to be inserted in the target description. If the query is supported by the match it is construed as the candidate inference of the analogy. A different approach is used by Falkenhainer's *contextual structure mapping* [3,5], which provides an elegant account of how to relax both the identicality and 1:1 constraints of structure-mapping when doing so provides more useful conjectures for the analogizer.

This paper describes a new technique, *pragmatic marking*, which is consistent with both standard and contextual structure mapping. The idea is to filter what subsets of local matches are

Figure 1: An example of SME input descriptions

We use this simple example for illustration only, realistic representations are typically much larger. The italicized numbers are not part of the representation, but have been introduced to provide a convenient means for refering to subexpressions later on.

Base domain:

```
1 (IMPLIES 2 (AND 3(SENSITIVE-TO 4 LITMUS32 5 ALCOHOL-VAPOR) 6 (INSIDE 7 COOLANT 8 SUMP)
                  10 (HELD-CLOSE LITMUS32 SUMP))
             11 (DETECTABLE 12 (GIVES-OFF COOLANT ALCOHOL-VAPOR)))
13 (IMPLIES 14 (LIQUID COOLANT) 15 (POSSIBLE (GIVES-OFF COOLANT ALCOHOL-VAPOR)))
16 (IMPLIES 17 (DECREASED 19 (PRESSURE SUMP))
          20 (INCREASED 21 (FLOW-RATE 22 (FLOW 23 STILL SUMP COOLANT 24 PIPE))))
26 (IMPLIES 27 (INCREASED (PRESSURE SUMP))
             28 (DECREASED (FLOW-RATE (FLOW STILL SUMP COOLANT PIPE))))
29 (IMPLIES 30 (DECREASED 31 (AREA PIPE)) (DECREASED (FLOW-RATE (FLOW STILL SUMP COOLANT PIPE))))
32 (IMPLIES 33 (INCREASED (AREA PIPE)) (INCREASED (FLOW-RATE (FLOW STILL SUMP COOLANT PIPE))))
33 (CAUSE 34 (GREATER 35 (PRESSURE STILL) (PRESSURE SUMP)) (FLOW STILL SUMP COOLANT PIPE))
36 (FLAT-TOP COOLANT)
```

Target domain:

```
1 (INCREASED 2 (FLOW-RATE 3 (FLOW 4 EFFLUENT 5 HEAT-SINK 6 HEAT 7 HX)))
8 (DETECTABLE 9 (GIVES-OFF EFFLUENT 10 RADIATION))
11 (CAUSE 12 (CONTAINS EFFLUENT 13 STRONGTIUM) (GIVES-OFF EFFLUENT RADIATION))
14 (LIQUID EFFLUENT)
15 (FLAT-TOP EFFLUENT)
16 (GREATER 17 (TEMPERATURE EFFLUENT) 18 (TEMPERATURE HEAT-SINK))
```

considered by whether or not they can support candidate inferences relevant to the analogizer's stated goal. Unlike ACME's query mechanism, this technique does not require the actual form of the candidate inference to be specified in advance. Thus our technique is better able to support the use of analogy in modeling problem-solving and discovery.

Section 2 reviews the SME algorithm using an example. Section 3 describes the *GreedyMerge* algorithm for efficiently combining local matches into consistent global interpretations. We analyze its theoretical properties and we demonstrate empirically that it tends to be optimal, in that the first interpretation it provides is usually the same as the best interpretation found by the exhaustive SME merge algorithm. Section 4 describes pragmatic marking, analyzes its complexity, and illustrates it by example. Finally, we discuss our plans for future work.

2 How SME works

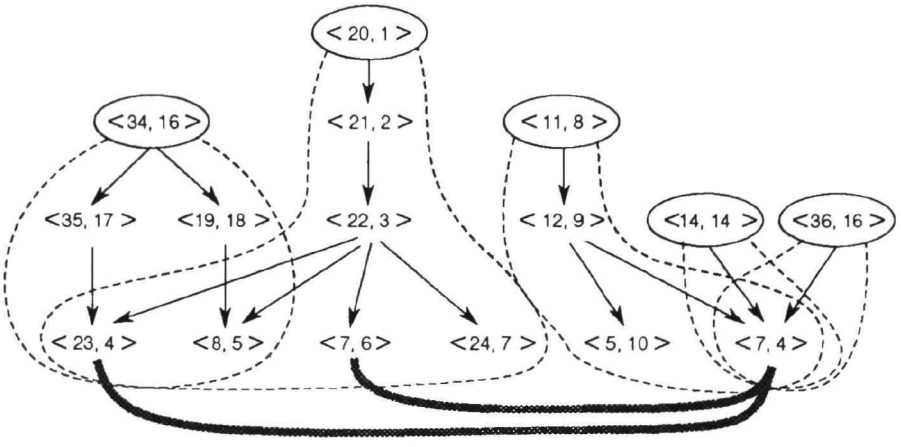
Here we sketch the standard SME algorithm to provide the backdrop for our improvements (see [7] for details). SME takes as input two propositional descriptions, a *base* and a *target*. It produces as output a set of global interpretations (gmaps) of the comparision. Each gmap contains a set of *correspondences* linking items in the base and target (including both entities and statements about them), a *structural evaluation score* which provides an indication of match quality, and a set of *candidate inferences*. The candidate inferences are statements in the base which can be hypothesized to hold in the target as a result of the gmap's correspondences. Each candidate inference is a surmise, and hence must be evaluated by other means to ensure its validity.

Figure 1 shows a simple example we use through the paper for clarity. Consider a case-based design system, which already had designed a still and was now working on a recycling plant. The base domain shows part of what it might retain about the still, and the target shows part of the description of the new design. This analogy can help solve two problems: how one might detect radiation in the effluent and how one might increase the rate of waste heat removal.

SME begins the mapping process by computing *match hypotheses* (MH's), each representing

Figure 2: Hypothesized local matches for the comparison

Each match hypothesis has the form < *Base,Target* >, where *Base* and *Target* are expression numbers from Figure 1. The roots of the graph are circled, and the pmap defined by each root is indicated by dotted lines. The thick lines indicate nogoods. Only structurally consistent MH's are shown for clarity.



a potential correspondence between an item of the base and an item of the target[1]. Figure 2 depicts the match hypotheses for our example. These local matches must be carefully filtered and combined to build structurally consistent interpretations. First, MH's involving items whose arguments cannot be placed in correspondence are eliminated from further consideration. In our example, the hypothesized match between these two statements

```
B: (CAUSE (GREATER (PRESSURE STILL) (PRESSURE SUMP)) (FLOW STILL SUMP COOLANT PIPE))
T: (CAUSE (CONTAINS EFFLUENT STRONGTIUM) (GIVES-OFF EFFLUENT RADIATION))
```

fails because neither of the corresponding arguments can match, while

```
B: (DETECTABLE (GIVES-OFF COOLANT ALCOHOL-VAPOR))
T: (DETECTABLE (GIVES-OFF EFFLUENT RADIATION))
```

is locally consistent, given the hypothesized pairings between COOLANT and EFFLUENT and between ALCOHOL-VAPOR and RADIATION. (These pairings can be considered as the *arguments* of the match hypothesis.) Next, SME installs local consistency constraints (*nogoods*) between pairs of MH's to mark potential violations of the 1:1 constraint. That is, the MH which maps COOLANT to EFFLUENT cannot ever be part of the same interpretation as the MH which maps COOLANT to HEAT. These local inconsistencies are propagated up the argument structure of the match hypotheses, to rule out MH's whose argument matches do not suggest consistent correspondences. Those MH's which remain become the grist for gmap construction.

Constructing maximal sets of MH's is the goal of gmap construction. A gmap is maximal if adding another MH causes structural inconsistancy. It is useful to view the set of match hypotheses as a partial order, with the MH's concerning object correspondences forming the bottom elements and inclusion relationships determined by the argument structure. Call an MH a *root* if it is consistent and is not an argument of some other match hypothesis. The roots of this graph are the initial gmap candidates, or pmaps, for "partial mappings" (Again, see Figure 2).

So far, the computational complexity is low. If n is the number of items in the base and target, then finding match hypotheses and local inconsistent combinations are both $\mathcal{O}(n^2)$, and the various propagation steps are $\mathcal{O}(log(n))$. Exhaustively combining pmaps into gmaps is the expensive part. It begins cheaply, by taking the union of the constraints for each pmap's correspondences to

[1]The rules which guide MH construction are programmable. To simulate structure-mapping, attributes and relational items must have identical functors. Different rule sets can be used to implement context-sensitive methods for relaxing identicality [5] and even simulate certain aspects of ACME[4].

compute what it is inconsistent with ($\mathcal{O}(n^2)$). The standard SME algorithm builds every possible complete gmap by making successive merges, subject to these consistency constraints, until no larger combinations can be built. If p is the number of pmaps, there are at worst $p!$ gmaps. This, of course, is expensive. Typical examples perform much better than this, due to the strong filtering effects of structural consistency. As [7] reports, on many complex examples SME takes only a few seconds of CPU time. However, we have found examples that can produce thousands of gmaps, and take days of CPU time to compute.

Finally, the structural evaluation and candidate inferences for each gmap are computed. These operations are of low complexity [7]. The structural evaluation score computation is irrelevant for this paper, see [7,8] for details. The only important feature is that the structural score of a gmap is the sum of its MH's scores, so it can easily be computed for pmaps and combined during merging.

Candidate inferences are computed by finding structure in the base which is consistent with a gmap's correspondences, but is not in fact included in them. Thinking now of the base domain as a graph, we are seeking structures which are roots (e.g., they are not themselves arguments of another item) and which have some, but not all, of their subitems mapped by the correspondences. Such items comprise potential new knowledge about the target, and are carried over by making the substitutions defined by the correspondences. Skolem functions are provided for base objects not mentioned in the correspondences. One candidate inference from a gmap resulting from our example comparison is:

```
(CAUSE (GREATER (TEMPERATURE EFFLUENT) (TEMPERATURE HEAT-SINK)) (FLOW EFFLUENT HEAT-SINK HEAT HX))
```

because the base structures *34* and *22* can map onto the target structures *16* and *3* respectively, while *33* in the base has no correspondence in the target (see Figures 1 and 2).

We believe the ability to generate structurally justifiable conjectures about the target is a central feature of analogy, responsible for its important role in creative problem solving and discovery (c.f. [2]). The rest of this paper shows how to achieve uniformly low complexity in gmap construction (at the cost of not always providing the optimum answer) and how to tune SME to produce novel candidate inferences relevant to the analogizer's goals.

3 Greed

The *greedy method* is a standard technique for combining a set of constrained, local solutions into a good global solution. The idea is that (a) finding a global solution can be modeled as deciding which local solutions to include and (b) some indication of "quality" exists for ordering local solutions[1]. Roughly, it works like this: Pick the best local solution. This rules out other choices, namely those which are inconsistent with the one picked. Throw away those which are inconsistent with your first choice. Now augment your solution with the best of the remaining local solutions. Again, this may rule out further choices, so one continues filtering and selecting until no more choices remain. The result is a single solution which is often, but not always, optimal.

The simplest version of *GreedyMerge* casts gmap construction as a sequence of decisions about which pmaps should be combined. The ordering is provided by the pmap's structural evaluation score. Starting with the largest, each pmap is merged into the solution under construction, unless doing so would violate structural consistency. If a pmap is inconsistent with the solution, it is skipped. By starting with the largest we improve our chances of getting the best solution.

The attraction of greedy methods is low complexity. Their drawbacks are (1) the solution may not be optimal and (2) obtaining useful alternative interpretations can be difficult. Whether or not the first problem is significant for natural representations is an empirical question addressed below. The second problem is very important. We view the ability to generate multiple interpretations of an analogy as critical. Even with a firm goal in mind, there can still be several ways to interpret an analogy (c.f. the Contras example in [12]).

There are several ways that multiple interpretations could be generated. One algorithm we explored generated an approximation of the top n gmaps based on their structural evaluation. This is often not a good strategy. Consider a very large base and a medium-sized target, so that many small, semi-independent pmaps are formed as well as several large ones. The gmaps for such comparisons can often be divided into several families of basically different interpretations, with each family member varying only in which small pmaps are included. In such cases the top n gmaps

Figure 3: Greedy Merge Algorithm
We assume that the standard SME algorithm has been executed up to the stage of constructing pmaps.

1. Place pmaps in descending order based on their structural evaluation score.
2. PMAPS ← the set of pmaps; USED ← { }
3. Repeat for desired number of interpretations
    3.1. MAPPING ← $\{\text{PMAP}_i\} \ni \text{PMAP}_i \notin \text{USED}$
        and $\forall j < i$ $\text{PMAP}_j \in \text{USED}$
    3.2. For each PMAP$\in$ PMAPS
        3.2.1. If PMAP is consistent with MAPPING Then
            3.2.1.1. MAPPING ← MAPPING $\cup$ PMAP
            3.2.1.2. USED ← USED $\cup$ {PMAP}
    3.3 Output MAPPING

Figure 4: Empirical Results of *GreedyMerge*

	Types Of Analogies		
	Object	Physical Systems	Stories
Number of matches	8	20	28
Min/Max number of gmaps	1/3	3/81	3/160
Min/Max GreedyMerge time (Sec.)	0/0.6	0.03/1.3	0.5/7
Min/Max FullMerge time (Sec.)	0/2.6	0.6/235	0.6/3335
Percentage of cases Greedy is optimal	100%	85%	96%
Lowest ratio Greedy Score / Best Score	100%	67%	91%

are likely to be trivial variations on the top theme, and since these will largely share the same set of candidate inferences, this is often undesirable. What we usually want is an alternate interpretation which is radically different. This suggests generating subsequent gmaps by starting with pmaps which are as large as possible but inconsistent with previously generated interpretations.

The algorithm we currently use (see Figure 3) starts by greedily generating the best gmap, and ensures that its gmaps are representative by always starting an alternate interpretation from a seed pmap which has never been used in any other interpretation. By adding the unused pmap first we ensure that we get a significantly different interpretation – it must be different since it contains a (hopefully large) structure which is inconsistent with all previously generated gmaps. Since the candidate inferences are based on the MHs these are also likely to be different. Note *GreedyMerge* reduces to the original greedy algorithm when generating only one interpretation. Each successive gmap starts with the largest unused pmap. All interpretations generated are maximal since *GreedyMerge* always attempts to add all pmaps to the interpretation during construction.

GreedyMerge is $\mathcal{O}(nlog(n))$ in the number of pmaps, and $\mathcal{O}(n)$ in the number of interpretations generated. The number of pmaps is $\mathcal{O}(n^2)$ in the size of the base and target, in worst case. In practice the number of pmaps tends to be much smaller, since only plausible MH's are generated, and these tend to cluster into reasonably large pmaps.

GreedyMerge has been tested on over fifty different analogies, ranging from comparisons between physical phenomon, short stories, and object descriptions, drawn from the library of SME examples. Figure 4 summarizes the results. The stories show the most dramatic speedup – one story could not be included because the exhaustive algorithm failed to terminate after several days of computation, yet *GreedyMerge* found a reasonable interpretation in under a minute. And in most cases the first gmap generated by *GreedyMerge* was identical to the best gmap found by the exhaustive merge algorithm.

Why does *GreedyMerge* do so well? Typically, these large examples have a few large pmaps, only some of which are mutually inconsistent, and a much larger set of small pmaps. Thus the first few decisions are the really critical ones, and they are relatively easy to make. When will *GreedyMerge* fail? There are two kinds of cases where it should do poorly. The first is when there are many large pmaps with a high degree of mutual inconsistency, since many more decisions have

to be correct, and hence the chance of error grows. This was the problem in the few cases (4 out of 56) where a non-optimal solution was generated. The second is when an initial, large pmap is inconsistent with every member of a large set of small but mutually compatible pmaps which in fact outweigh the initial one. We do not know how likely such situations are in natural representations. Fortunately, the ability to generate radically different interpretations provides a way to recover from such problems.

4 Pragmatism

The power of analogy comes from its ability to shed new light on the target by importing knowledge from the base. Retaining this ability using *GreedyMerge* requires modifying SME further. The reason is that the structurally best match may not always provide the most relevant inferences [3,5]. Returning to our example, the structurally best interpretation places the two flows in correspondence. But what if our goal is to propose how to detect strongtium in the recycler's effluent? As we find below, an interpretation which maps COOLANT to EFFLUENT is better for this purpose, even though a smaller structure is mapped. When using the original SME merge algorithm, one simply searches the interpretations to find a relevant inference. Since *GreedyMerge* is not exaustive, we must take care to ensure that relevant interpretations are actually generated.

Unfortunately, the techniques used by ACME provide no leverage here. Their techniques seem most useful for modeling instructional analogies, where a teacher may explicitly provide correspondences or point out which facts are most important. Here there is no correspondence involving the base fact that we wish to bring over, so it cannot be given extra weight or identified a priori as interesting. Introducing a query fact in the target does not help – if we knew the form of the query fact, we wouldn't need analogy to solve the problem. To get the novel inferences required for analogical problem solving requires a more generative solution.

Our *pragmatic marking* technique operates by looking for interpretations which can potentially import relevant base structure into the target. How can the relevant part of the base be found? Suppose we have target item G as our goal. That is, we want to find how G might legitimately be inferred on the basis of other (perhaps new) items in the target. For concreteness, suppose our goal is:

```
(DETECTABLE (GIVES-OFF EFFLUENT RADIATION))
```

Consider the set of match hypotheses generated for a comparison. The interpretations we are interested in must include a match hypothesis MH_G involving G, since only they can provide the structural grounds for candidate inferences involving G. (If G is not involved in any match hypotheses, then it cannot be the subject of any candidate inferences and hence we immediately know the comparison is useless for this purpose.) This means that the interpretation must in turn include MH's for the corresponding arguments of MH_G, and possibly for some larger structure of which it is a part.

Now consider the projection of MH_G onto the base domain. Again viewing the base as a graph, any pmap which includes the subgraph rooted in MH_G could provide inferences. However, pmaps which do not include this subgraph can also contribute to the structural grounding of an inference, so we must carefully examine them as well. There is some subset of roots of the base which contain MH_G's projection. Any pmap whose base projection lies outside this subset of the graph can be ignored, since it does not include the projection of our goal onto the base. Furthermore, any pmap inside this subset of the graph can be ignored if it is inconsistent with the correspondences implied by MH_G, since it could not be part of a gmap with it.

This intuitive picture provides the basis for the pragmatic marking algorithm (Figure 5). It is slightly more complicated to take into account the fact that there can be more than one MH_G, but otherwise is straightforward. The information required for the functions *TargetItem*, *BaseItem*, *Roots*, *BaseRoots*, *Descendants*, and *Nogood* is already computed in the process of generating pmaps. The complexity is thus $\mathcal{O}(|\{\text{pmap}\}| \times |\{MH_G\}|)$.

Figure 6 illustrates the results of two queries in our extended example. With the query about radiation detection, three out of the five pmaps are potentially relevant, and *GreedyMerge* successfully combines them all. The inference which results may be paraphrased as "By finding something which is sensitive to radiation, like litmus paper is sensitive to alcohol vapor, and holding it close to

Figure 5: Pragmatic Marking Algorithm

We assume that the standard SME algorithm has been executed, independent of any query, through pmap construction.

1. Let $\{MH_G\} = \{M | TargetItem(M) = G\}$
2. RELEVANT $\leftarrow$ {}
3. For each $MH_G \in \{MH_G\}$,
   - 3.1 For each pmap$_i \in$ pmaps
     - 3.1.1 If $Descendants(\text{pmap}_i) \cap Descendants(MH_G) \neq \{\}$ then go to 3.1.4
     - 3.1.2 If $BaseRoots(\text{pmap}_i) \cap Roots(BaseItem(MH_G)) = \{\}$ then skip.
     - 3.1.3 If $Nogood(\text{pmap}_i, MH_G)$ then skip.
     - 3.1.4 Otherwise, RELEVANT $\leftarrow$ RELEVANT $\cup$ pmap$_i$
4. *GreedyMerge*(RELEVANT)

Figure 6: Inferences generated in response to queries

```
G = (DETECTABLE (GIVES-OFF EFFLUENT RADIATION))

 There are 1 relevant interpretations:
GM1: 4 correspondences, SES = 2.5
  Object mappings: COOLANT <-> EFFLUENT, ALCOHOL-VAPOR <-> RADIATION
  Candidate Inferences:
(IMPLIES (AND (SENSITIVE-TO (:SKOLEM LITMUS32) RADIATION) (INSIDE EFFLUENT (:SKOLEM SUMP))
              (HELD-CLOSE (:SKOLEM LITMUS32) (:SKOLEM SUMP)))
         (DETECTABLE (GIVES-OFF EFFLUENT RADIATION)))

G = (INCREASED (FLOW-RATE (FLOW EFFLUENT HEAT-SINK HEAT HX)))

 There are 1 relevant interpretations:
GM5: 10 correspondences, SES = 4.375
  Object mappings: PIPE <-> HX, COOLANT <-> HEAT, STILL <-> EFFLUENT, SUMP <-> HEAT-SINK
  Candidate Inferences:
(CAUSE (GREATER (TEMPERATURE EFFLUENT) (TEMPERATURE HEAT-SINK)) (FLOW EFFLUENT HEAT-SINK HEAT HX))
(IMPLIES (INCREASED (AREA HX)) (INCREASED (FLOW-RATE (FLOW EFFLUENT HEAT-SINK HEAT HX))))
(IMPLIES (DECREASED (TEMPERATURE HEAT-SINK)) (INCREASED (FLOW-RATE (FLOW EFFLUENT HEAT-SINK HEAT HX))))
```

the effluent's container, one may detect when the effluent is giving off radiation." Notice that this interpretation is not the structurally best, which makes the flows correspond but is inconsistent with the mapping of COOLANT to EFFLUENT. The second question exploits the structurally larger interpretation, suggesting that in order to bring about an increase in the rate of heat removal, one can either increase the area of the heat exchanger HX or decrease the temperature of the heat sink. We have also successfully tested pragmatic marking on a variety of standard SME examples, with correct results in each case.

5 Discussion

We have seen how the SME algorithm can be modified to efficiently generate interpretations of analogies by using a greedy merging algorithm, and demonstrated that pragmatic marking can focus its efforts on just those interpretations likely to lead to relevant, novel candidate inferences. In moving from an exhaustive algorithm to a polynomial one we give up the guarentee of optimality, but as our empirical results indicate, we lose little by doing so.

There are several directions to explore next. For example, sometimes degrees of certainty or relevance can be estimated for items in a representation. It would be useful to exploit such information, as ACME does. Combining scores for certainty and relevance with the structural evaluation score used by *GreedyMerge* could provide an increased sensitivity to relevance that might be useful on larger problems. We also plan to use these techniques to embed SME into a larger simulation of human problem-solving activity.

6 Acknowledgements

This paper benefited from discussions with Dedre Gentner, Brian Falkenhainer, and Lenny Pitt. Gordon Skorstad helped draw figures, and Janice Skorstad provided valuable assistance with the SME

data. This research was supported by the Office of Naval Research, Contract No. N00014-85-K-0559, an NSF Presidential Young Investigator award, and an equipment grant from IBM.

References

[1] Aho, A., Hopcroft, J. and Ullman, J. *Data Structures and Algorithms*, Addison-Wesley, Reading, Massachusetts, 1985.

[2] Burstein, M. H. Incremental learning from multiple analogies. In A. Preiditis (Ed.), *Analogica: Proceedings of the First Workshop on Analogical Reasoning*, London, Pitman Publishing Co., 1988

[3] Falkenhainer, B., An examination of the third stage in the analogy process: Verification-Based Analogical Learning, Technical Report UIUCDCS-R-86-1302, Department of Computer Science, University of Illinois, October, 1986. A summary appears in *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, Milan, Italy, August, 1987.

[4] Falkenhainer, B., The SME user's manual, Technical Report UIUCDCS-R-88-1421, Department of Computer Science, University of Illinois, 1988.

[5] Falkenhainer, B. Contextual Structure Mapping. Unpublished manuscript.

[6] Falkenhainer, B., K.D. Forbus, D. Gentner, The Structure-Mapping Engine, *Proceedings of the Fifth National Conference on Artificial Intelligence*, August, 1986.

[7] Falkenhainer, B., Forbus, K., Gentner, D. The Structure-Mapping Engine: Algorithm and examples *Artificial Intelligence*, **41**, pp 1-63, 1989.

[8] Forbus, K. and Gentner, D. Structural evaluation of analogies: What counts? *Proceedings of the Cognitive Science Society*, August, 1989.

[9] Gentner, D., The structure of analogical models in science, BBN Tech. Report No. 4451, Cambridge, MA., Bolt Beranek and Newman Inc., 1980.

[10] Gentner, D., Structure-mapping: A theoretical framework for analogy, *Cognitive Science* **7**(2), 1983.

[11] Gentner, D., Mechanisms of analogical learning. To appear in S. Vosniadou and A. Ortony, (Eds.), *Similarity and analogical reasoning.* Presented in June, 1986.

[12] Holyoak, K. & Thagard, P. Analogical mapping by constraint satisfaction, *Cognitive Science* **13**, 295-355, 1989.

[13] Skorstad, J., Falkenhainer, B., Gentner, D., Analogical Processing: A simulation and empirical corroboration, in: *Proceedings of the Sixth National Conference on Artificial Intelligence*, Seattle, WA, August, 1987.

[14] D. Abstraction processes during concept learing: A structural view. In *Proceedings of the Tenth Annual Meeting of the Cognitive Science Society*, Montreal, August, 1988.

Analogical Interpretation in Context

Brian Falkenhainer
Xerox Palo Alto Research Center
3333 Coyote Hill Road, Palo Alto CA 94304

Abstract: This paper examines the principles underlying analogical similarity and describes three important limitations with traditional views. It describes *contextual structure-mapping*, a more knowledge intensive approach that addresses these limitations. The principle insight is that each element of an analogue description has an identifiable *role*, corresponding to the dependencies it satisfies or its relevant properties in the given context. Analyzing role information provides a powerful framework for characterizing analogical similarity, relaxing the one-to-one mapping restriction prevalent in computational treatments of analogy, and understanding how such similarities may be used to assist problem solving. Second, it provides a unifying view of some of the central intuitions behind a number of converging efforts in analogy research.

1 Introduction

The core of analogy is mapping: the identification of correspondences between two analogues and using them to adapt a prior experience to a new situation. To identify similarities, most approaches find matching patterns in the two analogue's representation form, seeking both isomorphic structures and features described by the same predicate.

One such system is SME [7], an analogical mapping program originally developed to study Gentner's (1983) Structure-Mapping theory. However, in using SME for complex problem solving tasks [4, 5], three fundamental problems were found with traditional views of analogy. First, the restriction that matching expressions must be represented by the same predicate is too strong, and existing solutions based on conceptual closeness (e.g., ISA hierarchies) are too weak. Second, the one-to-one mapping restriction is too strong whenever multiple functions are supported by a single element in one of the analogues (e.g., function sharing). Third, the correspondences cannot always be determined from the initial descriptions of each analogue since their descriptions may not contain all of the relevant objects or inferences. Thus, additional information may have to be inferred or retrieved during the mapping computation.

This paper describes *contextual structure-mapping* (CSM), an approach that relaxes these restrictions while ensuring meaningful similarity judgments and tractable computation. It was essential to the success of PHINEAS [4, 5], a program that constructs new causal models of observed phenomena based on their similarity to understood phenomena. The key idea is explicit consideration of the contextual factors affecting analogical interpretation. Specifically, each element of an analogue description has an identifiable *role*, corresponding to the dependencies it satisfies or its relevant properties in the given context. The notion of role makes two important contributions. First, it provides a powerful framework for characterizing predicate "similarity", relaxing the one-to-one mapping restriction, and focusing problem solving and memory retrieval aimed at elaborating analogue correspondences. Second, it provides a unifying view of some of the intuitions behind derivational replay [3, 11], tweaking [10], knowledge-based pattern matching [1].

We begin by briefly describing role information. We then describe how this information is used to compute similarity correspondences within SME and adapt a prior problem solving experience to a new case. Finally, we discuss how the concepts presented here unify and explain several aspects of related work.

2 The role of role

We assume that each analogue is described by a set of *expressions*, where an expression may be a predicate-calculus formula, a feature in a feature vector representation, or a node or link in a semantic network. Take $\mathcal{B}$ and $\mathcal{T}$ to denote the sets of expressions representing the base and target analogues, respectively. Take $\mathcal{K}$ to be one's complete body of knowledge, such that $\mathcal{B}, \mathcal{T} \subset \mathcal{K}$. In general, $\mathcal{K}$ will include or entail information about each analogue not explicit in $\mathcal{B}$ or $\mathcal{T}$. The general mapping goal is to find a set of correspondences between the elements of $\mathcal{B}$ and $\mathcal{T}$. Some correspondences may be directly identifiable from the base and target descriptions; others arise as a side effect of adapting base information to apply to the target case.

Most accounts of analogy and case-based reasoning establish similarity correspondences by testing for predicate or feature identicality, but this is too restrictive. The typical solution for allowing non-identical relations to match is to evaluate similarity by measuring conceptual closeness, using ISA hierarchies [13, 2] or a-priori similarity scores [9]. However, these approaches can produce unmotivated and incorrect similarity correspondences. They fail to recognize an important point: similarity is context sensitive. Having some aspects in common is not an explanation for why two predicates should be viewed as corresponding; only some of their properties are relevant for determining similarity in a given context.

For example, consider classifying a cylindrical tin cup with a handle as a `Hot-Cup` (adapted from [13, 11]). In general, a `Hot-Cup` is something that a person can lift and drink from while it holds a hot liquid. From a previous styrofoam cup (`scup`) example, the following sufficient conditions were found:

$$\texttt{Styrofoam(scup)} \wedge \texttt{Open-Conical(scup)} \wedge \cdots \rightarrow \texttt{Hot-Cup(scup)}$$

In classifying the tin cup via analogy to the styrofoam case, conceptual closeness measures might pair `Open-Conical` with the tin cup's `Open-Cylinder` (both open concavities) and `Styrofoam` to `Tin` (both materials). Yet, if the original dependencies satisfied by the `Styrofoam` condition are retrieved (or reconstructed), it becomes clear that this misses the point of the analogy: the aspect of styrofoam important in this context is its insulating characteristics, just as the tin cup's handle is important because it provides another form of insulation. In this context, the property styrofoam should map to the property of having a handle.

Most matchers require one-to-one mappings because allowing many-to-many mappings can dramatically increase the number of possibilities and lead to incoherence. However, in the styrofoam cup, styrofoam and conical shape provide insulation and a grasping area, respectively, while both functions are provided by the tin cup's handle. Here, an isomorphic mapping fails to fully capture the correspondence; a many-to-one mapping from { `Styrofoam`, `Open-Conical`} to { `Has-Handle`} is needed. Similarly, consider the dual murderer / victim roles of someone committing suicide. Due to function sharing, many-to-many mappings occur in many physical systems.

The key to relaxing these constraints and determining an expression's relevant aspects is an understanding of its *role* in the analogue description. In CSM, an expression's roles are identified by the dependencies it satisfies. Within each analogue, if $\mathcal{C}$ is some predication whose truth is dependent on predication $\mathcal{A}$, then the *role* of $\mathcal{A}$ is to satisfy the dependency relationship with $\mathcal{C}$. $\mathcal{A}$ may fill other roles as well, in as much as $\mathcal{A}$ satisfies other dependencies.

Correspondences between two analogues' expressions are determined by analyzing their roles in the context of their respective analogues. Thus, if the dependencies supported by an expression $\mathcal{E}_b$ may be satisfied in an alternate manner, say by expression $\mathcal{E}_t$, then $\mathcal{E}_b$ and $\mathcal{E}_t$ are considered *functionally analogous* and may be placed in correspondence. We may generalize this (and form a recursive definition) by saying that given two corresponding roles (i.e., not necessarily identical), their role fillers may be considered functionally analogous and eligible for being placed in correspondence, independent of predicate (or feature) identicality. Importantly, a fundamental component

of analogy is knowing what aspects of a given relation are relevant and hence be able to recognize alternate ways to achieve similar functionality. For example, the property **RAINFALL** should map to the property **IRRIGATION** if the role of these conditions is to ensure that a given crop receives sufficient water. Note they are not analogous in other roles, such as washing a plant's leaves. Further, to the extent that knowledge of $\mathcal{E}_b$'s role is incomplete and $\mathcal{E}_t$ does not provide identical functionality, the functionally analogous relation is merely plausible (e.g., the humidity associated with rainfall may be relevant).

Roles take many forms. In design, the role of particular design decisions and artifact components is the satisfaction of particular design specifications and rationale. The role of an agent's actions (as in planning or a story) may be in support of certain outcomes. In physical systems, the role of a given component is typically the behavior it contributes to. In deductive proof, the role of an antecedent is to provide logical support for its consequent. There are several ways to determine and exploit the role an expression is servicing. We discuss two of these next.

2.1 Explicit dependencies

When an expression's role is explicit in a given analogue representation, role determination consists simply of consulting this information. This appears in two forms. First, the roles of b_i and t_j may be explicit in both base and target descriptions. For example, expressions `P` and `R` will be placed in correspondence when matching `IMPLIES(P,Q)` with `IMPLIES(R,Q)`, since their respective roles are to deductively support `Q`.

Second, the role of b_i may be explicit in $\mathcal{B}$ but there is not enough information in $\mathcal{T}$ to identify its correspondent. In this case, the task is to take the unmapped b_i's role and search for a corresponding role and role filler applicable to the target setting. Since $\mathcal{T}$ is generally a subset of all available knowledge about the target, this case will require memory retrieval and inferencing to find the desired information. For example, mapping

$\mathcal{B}$: `High-Rainfall(region`$_1$`)` $\rightarrow$ `Well-Watered(region`$_1$`)`
to
$\mathcal{T}$: `Irrigated(region`$_2$`)`, `Arid(region`$_2$`)`, `Northern-Hemisphere(region`$_2$`)`

requires retrieval of

`Irrigated(region`$_2$`)` $\rightarrow$ `Well-Watered(region`$_2$`)`

in order to determine which expression in $\mathcal{T}$ corresponds to `High-Rainfall` in $\mathcal{B}$.

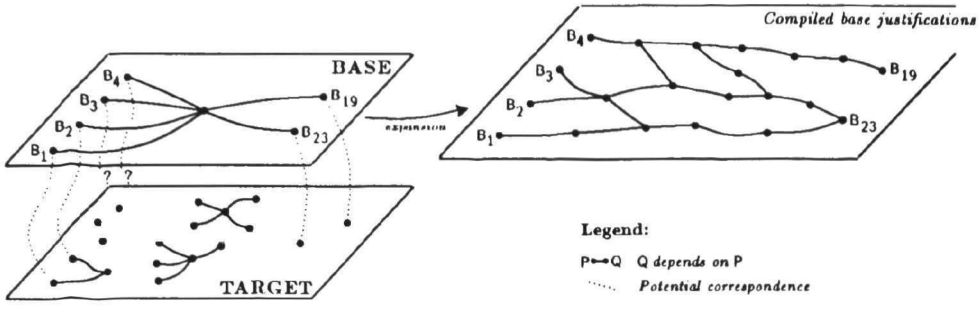
2.2 Compiled and abstracted knowledge

Representations that are adequate for traditional approaches to problem solving may not be suitable for performing analogical reasoning. AI systems tend to represent a minimalist approach to encoding knowledge, in which detailed descriptions or intermediate reasoning steps are avoided to promote efficiency of use. For example, a physical process may be modeled at some level of abstraction and a design plan may not contain the rationale behind the decisions it embodies. Indeed, knowledge compilation is a central goal of explanation-based generalization. While effective for accelerating reasoning, a great deal of information is intentionally removed. This poses a significant problem for adaptation of a given base analogue: these intermediate or second-order justifications are often needed to ascertain the requisite role information (Figure 1).

For example, consider the following schema used to explain why crops grow in `region`$_1$:

```
Cro p-Growing(region1)
   PRECONDITIONS   High-Rainfall(region1) ∧  Fertile-Soil(region1) ∧
                   Sunny(region1) ∧ ...
   EFFECTS         Growing-Crops(region1)
```

Figure 1: Compiled knowledge problem. Intermediate reasoning steps removed to increase problem solving efficiency are often required when considering how to elaborate analogical correspondences and adapt a solution to unanticipated cases. The **BASE** representation depicts a macro compiled from the inferences to the right. A question mark indicates a relevant base expression having no known target correspondent without more information about its role in the base context.



Why is the `High-Rainfall` precondition there? Suppose a deeper explanation is retrievable and reveals that this condition is important in this context because it ensures that the region is well watered. Without this deeper explanation, it would be impossible to justify why it is reasonable to place `High-Rainfall(region`$_1$`)` in correspondence with `Irrigated(region`$_2$`)`.

To address the compiled knowledge problem, we assume the availability of needed background knowledge, either cached or reconstructable. This background knowledge serves to decompose and elaborate the reasons underlying a particular dependency relationship. This information is consulted as needed during the mapping computation.[1] In the implementation, a `CACHE` field accompanies all compiled rules. For example, the crop growing schema should have an added `CACHE` field to store compiled reasoning steps like `High-Rainfall(region`$_1$`)` $\rightarrow$ `Well-Watered(region`$_1$`)`.

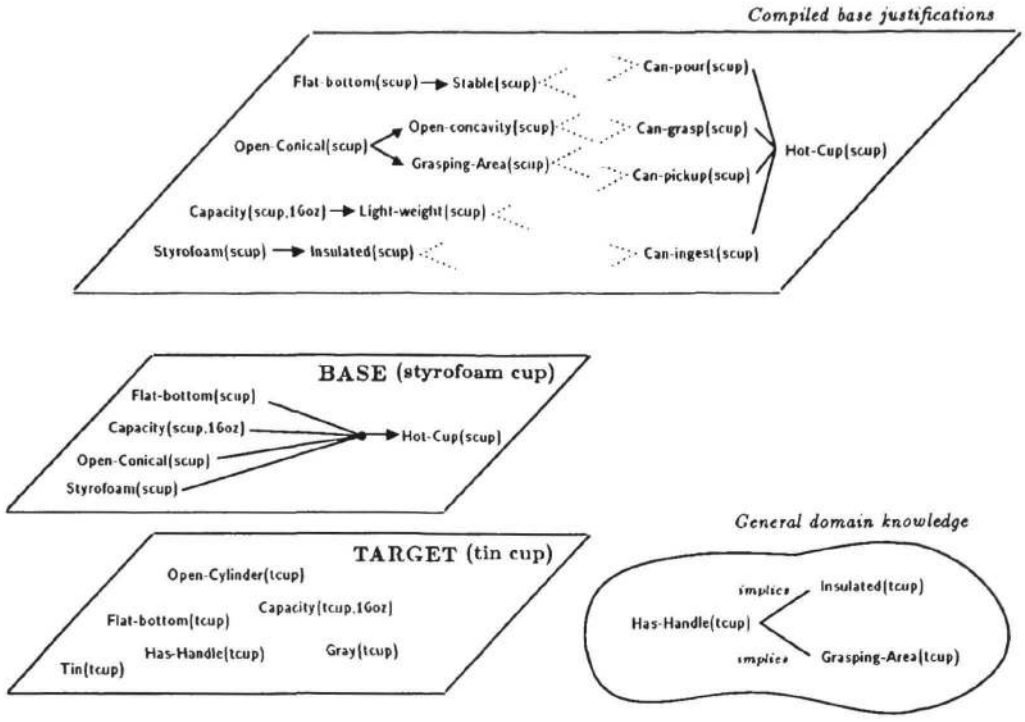
3 The Map and Analyze Process

The mapping process is traditionally depicted as a form of pattern matching between base and target descriptions. However, in realistic memories, mapping will be operating on a subset of all that is inferrably known about the base and target. Thus, the process of elaborating correspondences and adapting elements of the base to fit the target situation often requires inferring additional information during the mapping computation in response to mapping impasses. Rather than endow the mapping mechanism with unlimited inferencing power, we decompose the process to form a *map and analyze* cycle: use simple matching criteria to determine the best, initial mapping between the analogues, analyze the results and seek additional relevant information about unmatched areas, reexamine the mapping to determine the information's impact on the mapping (i.e., extensions or complete shifts), analyze the new results, etc.

The mapping phase is computed by SME [7] a general mapping tool which formulates mappings based on user-supplied *match rules*. Match rules specify which local, pairwise correspondences (called *match hypotheses*) between expressions and entities are possible, restrictions on how they may be combined, and preference criteria for scoring these combinations. Using the rules of contextual structure-mapping forms SME$_{CSM}$. Each time a mapping is computed, one out of the set of possible mappings is selected based on systematicity [8, 7] and relevance to the current problem solving goals [5, 6].

[1] Including all background knowledge in the original base and target descriptions would be too expensive. Additionally, which added details are needed cannot be identified until impasses arise during the mapping computation.

Figure 2: Categorizing a tin cup by analogy to a styrofoam cup.



In the remainder of this section, we illustrate how role information is used to provide controlled relaxation of the same predicate and one-to-one mapping restrictions.

3.1 Relaxing identicality

Given base and target expressions, what criteria are used to propose a match between them? We start with the standard set taken from Gentner's structure-mapping theory [7]. The main rule pairs expressions that use the same predicate. Additional rules are used to support objects and commutative predicates. However, these rules suffer from a dependence on identicality to initiate matches. To motivate matches between expressions using different predicates, we add a rule to support the functionally analogous criterion:

Rule 1 (Functionally Analogous) *Two expressions are considered* functionally analogous *and may match if they fill corresponding roles in the context of the structures being matched.*

When role information is explicit in the base and target representations, SME_{CSM} uses this rule to match functionally analogous expressions. When a relevant base expression has no discernible correspondent in the target case, its role is analyzed in greater depth.

To illustrate these match criteria, we use a simple example taken from [13, 11] and adapted to illustrate the key ideas. The task is to classify a given tin cup as an instance of `Hot-Cup`. The base exemplar is a previously classified styrofoam cup. Their descriptions are shown in Figure 2.

The process begins by comparing the initial descriptions of the two cases. `SME` finds that the relevant `Flat-bottom` and `Capacity` properties are shared by both cups. It also fails to find correspondents for the styrofoam cup's other important properties: `Open-Conical` and `Styrofoam`. During the analysis phase, more detail is retrieved about the roles of these two properties in

the styrofoam case's classification as a `Hot-Cup`. For example, its `Styrofoam` property provides insulation. The system then seeks aspects of the tin cup that fill these roles. For example, the tin cup's `Has-handle` property also provides insulation. When the mapping is reexamined, this added information now enables a complete match and successful classification of the tin cup as an instance of `Hot-Cup`.

3.2 Relaxing one-to-one

The notion of role provides exactly the right constraint for introducing many to many mappings while maintaining coherence and tractability. In CSM, the one-to-one restriction may be violated by a single base or target item filling multiple roles that are filled by multiple items in the other domain. In the cups example, Figure 2 shows the dependencies that motivate a many-to-one mapping from `Styrofoam(scup)` and `Open-Conical(scup)` to `Has-Handle(tcup)`.

The following three rules define *sanctioned* many-to-one mappings and enforce one-to-one mappings as the normal default by examining all cases of two base items mapping to a single target item.

Rule 2 (Direct role-filler) *Multiple base items b_1 and b_2, filling roles $\mathcal{R}_{b1}$ and $\mathcal{R}_{b2}$ respectively, may map to a single target item, t_i, filling roles $\mathcal{R}_{t1}$ and $\mathcal{R}_{t2}$, if role $\mathcal{R}_{b1}$ corresponds to role $\mathcal{R}_{t1}$ and role $\mathcal{R}_{b2}$ corresponds to role $\mathcal{R}_{t2}$.*

This rule sanctions match hypotheses that may violate the one-to-one restriction. In the cups example, it pairs the `Has-Handle` predicate in the tin cup description to both `Styrofoam` and `Open-Conical`. However, unless this sanctioning is propagated to their respective subexpressions (e.g., the predicates' arguments), a one-to-one restriction will still be in effect.

Rule 3 (Role-filler sub-expressions) *Multiple base items may map to a single target item if the base and target items are subexpressions of a sanctioned many-to-one mapping.*

In the cups example, because `Styrofoam` and `Open-Conical` apply to the same object, `scup`, only a one-to-one mapping from `scup` to `tcup` is needed and this rule does not apply.

With sanctioned violations of one-to-one identified, we are now ready to define when two match hypotheses are conflicting. Due to its non-monotonic nature, implicit in the following rule is the assumption that all sanctioned pairings are known at the time the rule is invoked.

Rule 4 (One-To-One) *Unless explicitly sanctioned, two match hypotheses are* conflicting *if they pair multiple base items to the same target item.*

Thus, the two match hypotheses (`Styrofoam`, `Has-Handle`) and (`Open-Conical`, `Has-Handle`) are compatible in the cups example. A symmetric set of three rules exist for multiple target items mapping to a single base item. Together, the six rules identify sanctioned many-to-many mappings and enforce one-to-one for all other, unsanctioned cases.

4 A Unifying View

The contextual structure-mapping framework provides a unifying view of the basic intuitions behind several recently developed methods for computing similarity. *Derivational replay* mechanisms make the observation that it is typically easier to reuse the problem solving process of a prior episode than the episode's final solution [3]. Taking a stored problem solving plan, a new problem is solved by replaying the plan top-down, resolving subgoals that no longer apply in the current situation. CSM's definition of *functionally analogous* explains the underlying intuition behind replay's appeal

over mapping only a final solution: the root problem in reusing a prior solution is the need to understand the various roles or functions the solution fulfills. In problem solving, adapting a prior solution to a new problem instance is greatly simplified if decisions' rationale are known, so that their intent can be satisfied without necessarily adhering to the same decisions. Top-down replay achieves this by essentially reseeking each role filler in the new situation. An alternate method would start at the solution, and work backwards, analyzing role information at solution transfer impasses. The implicit assumption in replay mechanisms is that it is more efficient to work forward, replaying the entire decision-making process, than to work backward, reconsidering the decisions (alternate ways to achieve functionality) where needed. This tradeoff is influenced in part by the solution's modularity.

The Yale SWALE project [10], Kedar-Cabelli's PER [11], and PHINEAS (which uses CSM) [4, 5] are efforts to use analogy to reduce the cost and increase the creativity of explanation building. These systems demonstrate operations very closely related to the notions of role and derivational replay. For example, SWALE applies stored *explanation patterns* (schemas) to new situations, *tweaking* them as needed to adapt to the situation's novel aspects. Like Kedar-Cabelli's PER model, it attempts to rederive portions of a prior explanation that do not apply in the new situation. For the tweaking operation, SWALE uses a set of revision rules, such as *substitute alternate theme* or *substitute related action*. These suggest ways to repair inapplicable portions of a recalled schema. CSM offers a more general explanation of these tweaking operations: determine the anomalous item's role and seek elements of the current situation that could satisfy that role's dependencies. For example, if the current actor lacks a requisite theme, *substitute alternate theme* tries to find an alternate theme for the actor. In the CSM framework, this corresponds to an unsatisfied dependency (i.e., role) which either must be satisfied, assumed, or conjectured with new concepts.

PROTOS [1] performs diagnosis by relating a new case to a store of previously classified exemplars, performing classification by finding the exemplar that best matches the new instance. *Knowledge-based pattern matching* computes a match between an exemplar and a new instance by using the domain knowledge stored with each exemplar to explain the equivalence of non-identical features. For example, in comparing two chairs, LEGS and PEDESTAL are equivalent because they both provide SEAT-SUPPORT. In CSM terms, this process corresponds to showing that two features are functionally analogous. However, PROTOS limits the scope of a match to the explicitly given features of the two cases. It does not include the possibility that additional features may be derived or retrieved from memory in response to the needs of the match elaboration process. Furthermore, it uses a feature-vector representation (i.e., a set of attribute-value pairs) which is inadequate for representing a complex set of interrelationships between an analogue's parts.

5 Discussion

Analogical mapping for simple representations and tasks, particularly for within-domain comparisons, is a straightforward process. On the other hand, rich representations and complex problem solving tasks (both within and across domains) require a more sophisticated mechanism for computing similarities that is able to compare syntactically different analogues and select the best mapping when ambiguities arise. The utility of contextual structure-mapping for such settings has been demonstrated in PHINEAS [4, 5], a program that proposes qualitative causal explanations of observed, time-varying phenomena based on their similarity to understood phenomena. PHINEAS has been extensively tested on over a dozen examples, such as explaining evaporation by analogy to dissolving, heat flow and osmosis by analogy to liquid flow, and a variety of mechanical and electrical harmonic oscillators.

This paper has claimed that analogy requires role information to at least plausibly suggest the relevance and interrelatedness of each analogue's features. The ability to learn this relevance information is of fundamental importance. One approach is to use a developing analogy to motivate

specific questions about the world and use directed experimentation to answer them and ascertain the requisite relevance information [5]. A second approach is to again use a developing analogy to motivate specific questions, but place the system in a learning apprentice setting and obtain requisite relevance information from the user [1]. We are currently investigating a third approach: use inductive methods to suggest which factors are relevant to the concept under study.

Acknowledgements

Ken Forbus, Dedre Gentner, Danny Bobrow, and Mark Shirley provided insightfull discussions and helpful comments on prior drafts of this paper. The foundation for this work is taken from the author's dissertation, which was supported by an IBM Graduate Fellowship and by the Office of Naval Research, Contract No. N00014-85-0559.

References

[1] Bareiss, R., Porter, B., & Wier, C. (1987). Protos: An exemplar-based learning apprentice. *Proceedings of the Fourth International Workshop on Machine Learning.*

[2] Burstein, M. (1986). Concept formation by incremental analogical reasoning and debugging. In R.S. Michalski, J. Carbonell, T. Mitchell (Eds.), *Machine Learning: An Artificial Intelligence Approach, Volume II.*

[3] Carbonell, J. (1986). Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In R.S. Michalski, J. Carbonell, T. Mitchell (Eds.), *Machine Learning: An Artificial Intelligence Approach, Volume II.*

[4] Falkenhainer, B. (1986). *An examination of the third stage in the analogy process: Verification-based analogical learning* (Technical Report UIUCDCS-R-86-1302). Department of Computer Science, University of Illinois. A summary appears in *IJCAI-87.*

[5] Falkenhainer, B. (1988). *Learning from physical analogies: A study in analogy and the explanation process.* PhD thesis, University of Illinois, 1988.

[6] Falkenhainer, B. (1990). Contextual structure-mapping. (*submitted for publication*).

[7] Falkenhainer, B., Forbus, K.D., & Gentner, D. (1989). The structure-mapping engine: Algorithm and examples. *Artificial Intelligence, 41,* 1-63.

[8] Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science, 7.*

[9] Holyoak, K. & Thagard, P (1989). Analogical mapping by constraint satisfaction. *Cognitive Science, 13*(3).

[10] Kass, A., Leake, D., & Owens, C. (1986). SWALE, A program that explains. In R Schank (Ed.), *Explanation patterns: Understanding mechanically and creatively.* Lawrence Erlbaum Associates.

[11] Kedar-Cabelli, S. (1988). Formulating concepts and analogies according to purpose. PhD Thesis, Rutgers University.

[12] Kolodner, J., Simpson, R. & Sycara-Cyranski, K. (1985). A process model of cased-based reasoning in problem solving. *Proceedings of IJCAI-85.*

[13] Winston, P., Binford, T, Katz, B. & Lowry, M. (1980). Learning physical descriptions from functional definitions, examples, and precedents. *Proceedings AAAI-83.*

MULTIPLE ABSTRACTED REPRESENTATIONS IN PROBLEM SOLVING AND DISCOVERY IN PHYSICS

Nancy J. Nersessian
Program in History of Science
Princeton University

James G. Greeno
Stanford University and
Institute for Research on Learning

ABSTRACT

We discuss the process of mathematization in science, focussing on uses that theorists make of physical representations that we refer to as abstracted models. We review abstracted models constructed by Faraday and Maxwell in the mathematization of electromagnetic phenomena, including Maxwell's use of an analogy between continuum dynamics and electromagnetism. We discuss ways in which this example requires major modifications of current cognitive theories of analogical reasoning and scientific induction, especially in the need to understand the use of abstracted models containing theoretically meaningful objects that can be manipulated and modified in the development of new concepts and mathematized representations.

INTRODUCTION

Problem solving and discovery in physics involve the application of appropriate mathematical formalisms to specific configurations of physical phenomena. In the history of science, the process of figuring out the fit between mathematics and the physical world has been called "mathematization" (Koyre). That process consists of grasping the relational structure underlying phenomena; abstracting that structure; expressing the abstraction in mathematical formulae; and applying the formalism back to a wider class of phenomena. For example, "mathematization" of the motion of projectiles required formulating a relationship between the component forces acting on the body by abstracting the motion of an object in air to that of an idealized point mass in an empty three-dimensional, homogeneous, infinite space; and, as ultimately formulated in Newton's laws, this analysis showed the motions of projectiles, planets, and pendula to have the same formal structure.

Historically, the process of mathematization has often had as a central component the construction of a physical representation -- a model, a schematic representation, a diagram, etc. -- embodying tentative assumptions about the structural relationships under investigation. These representations are like equations and other descriptions that are physical objects in their own rights, enabling reflection and investigation of their properties separated from the phenomena they are meant to represent. We use the term "abstracted models" to refer to such models, schematic representations, diagrams, etc. Abstracted models allow scientists to manipulate familiar structures, observe consequences of adjusting relationships to satisfy domain constraints, and, ultimately, to generate equations that express the assumed relationships in

formal terms. Because abstracted models reify the hypotheses of the investigator, he or she can reason with and about the abstraction rather than the represented phenomena.

One advantage of reasoning with a model is that reasoning about the represented phenomena in all their complexity can create a cognitive "overload," since what is or isn't relevant is often not evident. Another advantage is that the abstracted model provides support for productive situated reasoning about interactions among hypothesized properties and relations. The objects in a model representation that correspond to properties and relations in the phenomena can be examined in different arrangements to allow exploration of interactions that cannot be produced and observed as directly in the domain of the phenomena. (For a discussion, see Greeno, 1989.) This paper is particularly concerned with an example of discovery that came from formulating and modifying a causal model with objects corresponding to properties and relations in phenomena of electricity and magnetism.

MATHEMATIZING ELECTROMAGNETIC PHENOMENA

The history of science is replete with examples. The derivation of the electromagnetic field equations provides a particularly salient case study. The standard textbook account at both the undergraduate and graduate levels (e.g., Jackson, 1962; Feynman *et al.*, 1963; Panofsky & Phillips, 1962) all present Maxwell as starting from a set of field equations for closed circuits plus the equation for continuity of charge shown in Table 1. Maxwell's problem is portrayed as that of reconciling these equations for the case of open circuits. According to this account, considerations of formal consistency required that he add a term to Ampere's Law to represent the contribution of electrostatic polarization to current.

Table 1 about here

Space limitations require that we present an exceedingly compressed version of the actual process through which the field equations were derived. Nersessian (1984, 1986, in press) has presented fuller analyses. What we hope to do here is to show how different the actual process was from that presented in textbooks and, in particular, to demonstrate how important various abstracted models were in the mathematization process.

Figure 1 about here

It was Maxwell who generated the field equations for electromagnetism, but his analysis of the problem began with specific abstracted models created by Faraday. In opposition to the mathematical representation of electric and magnetic actions as Newtonian actions at a distance by Ampère, Faraday hypothesized that the lines of force that form when iron filings are sprinkled around magnets and charged matter indicate that some real physical process is going on in the space surrounding these objects and that this process is part of the transmission of the actions. Figure 1(a) shows the actual lines as they form

around a magnet and Figure 1(b) shows an abstracted representation of these lines in geometrical and dynamical form. That this abstraction played a central role in his reasoning about electric and magnetic phenomena can be seen in the many line-like features that he incorporated into his descriptions of the actions and that guided his attempts to detect them experimentally. For our purposes, it is most notable that the only quantitative measure he introduced is between the number of lines cut and the intensity of the induced force. This relationship is incorrect, because "number of" is an integer while "field intensity" is a continuous function. The "mistake" occurs because in the abstracted model the lines are taken as discrete objects, while they actually spiral indefinitely in a closed volume.

Figure 2 about here

Near the end of his research, Faraday introduced another abstracted model representing the dynamical balance between electricity and magnetism. Figure 2(a) is Faraday's actual abstracted model. Figure 2(b) shows how the picture of interlocking curves is abstracted from the earlier abstracted model involving lines of force. For example, a lateral repulsion of the magnetic lines (outer lines) has the same effect as a longitudinal expansion of the current lines (inner lines).

Maxwell used both of Faraday's abstracted models in his first attempt to mathematize electromagnetism (Maxwell, 1890, pp. 155-229). In this analysis he replaced Faraday's relationship between the number of lines cut and the intensity of the induced force with a continuous measure by representing the lines of force as the flow of an incompressible fluid through fine tubes of variable section, filling all space. The interlocking curves, called "mutually embracing curves" by Maxwell (p. 194), formed the basis of his reciprocal dynamics. The effect of this abstraction on his thinking can be seen most directly in his complicated use of two fields -- one for a longitudinal measure of force and one for a lateral measure -- where we would now only use one.

Figure 3 about here

The abstractions that Maxwell took from Faraday were useful primarily for Maxwell's kinematical analysis. A dynamical analysis of the underlying forces that could produce the lines required the construction of a quite different abstracted model: one that would embody the dynamical relations between electric and magnetic forces. The abstraction that Maxwell constructed is an analogy between electromagnetism and continuum mechanics (fluids, elastic media, etc.). Maxwell first constructed a primitive abstracted model, shown in Figure 3(a), consistent with a set of constraints: a fluid medium composed of elastic vortices and under stress. With this form of the abstraction he was able to provide a mathematical representation for various magnetic phenomena. Analyzing the relations between current and magnetism required alteration of this abstraction. In Figure 3(a) all the vortices are rotating in the same direction, which means that if they touch, they will stop. Mechanical consistency, thus, requires the introduction of "idle wheels" surrounding the vortices, and Maxwell

argued that their translational motion could be used to represent electricity. Figure 3(b) shows a cross section of this altered abstracted model. For purposes of calculation, Maxwell had to make the elastic vortices into rigid pseudospheres. He next formulated the mathematical relations between currents and magnetism. It then took him nine months to figure out how to represent the final (and most critical) piece of the problem: electrostatic actions. He found that if he made the vortices elastic once again, and identified electrostatic polarization with elastic displacement, it was possible to calculate the wave of distortion produced by polarization. That is, adding elasticity to the abstraction enabled him to show that electromagnetic actions are propagated with a time delay, i.e., they are field actions and not Newtonian actions at a distance. At this point Maxwell had achieved a fully mathematized representation of the electromagnetic field.

DISCUSSION

The main point of this example is that it was through a process of embodying the structural relations between electric and magnetic actions in a series of abstracted models, reasoning with and about these, and manipulating them in various ways, that Maxwell generated the field equations for electromagnetism. Considerations of formal consistency, as presented in textbooks, played no significant role in this analysis. At the same time, the known mathematical structure of continuum dynamics motivated the analysis and provided the basis for achieving the goal of a mathematized representation of the dynamics of electromagnetic fields.

The analysis illustrates reasoning that is concretely situated, yet relies on a process of abstraction that is crucial to the success of the reasoning effort. The construction of an abstracted model removes a set of properties and relations from its initial context, creating objects in a new situation that can be analyzed and manipulated. Analysis of the properties of the abstracted model is situated in the context that the model provides. The effort to use that model as a representation for a different domain proceeds by considering various requirements that arise from features of the second domain that may be known or emerge in the problem-solving process. These requirements act as constraints on the model, as the theorist seeks a modified model that behaves in accordance with them. Modifications are sought with the condition that the modified model should have a mathematical structure that can be understood and expressed in formal terms.

The process of arriving at Maxwell's equations according to this analysis is significantly different from the processes of formula induction of the kind that Simon and his colleagues have studied (Langley, Simon, Bradshaw, & Zitkow, 1987). That process is situated in a context of numerical data and symbolic expressions that specify numerical operations between variables. The search takes place primarily in a space of formulas, with theoretical entities introduced as needed to account for invariant relations among numbers. In Nersessian's interpretation of Maxwell's discovery, the search is primarily a search for a model with a coherent causal structure that originates in one domain but is

changed so that its behavior fits constraints that hold in a different domain. After the change, of course, it no longer provides an accurate representation of the phenomena in the domain from which it was abstracted.

Analogical reasoning of the kind identified in this analysis also differs significantly from the analyses that have been studied by cognitive psychologists such as Gentner (1983), by Gick and Holyoak (1983; 1980). In these analyses, an attempt is made to match the relational structures of two or more domains, with those analogies considered best that provide the closest match between patterns of relations. In contrast, the analogy between continuum dynamics and electromagnetic fields was productive because of the possibility of constructing an abstracted model of continuum dynamics that could be changed to fit the constraints of electromagnetism.

The success of the analogy depended upon having a representation that could be analyzed and manipulated as an object separate from its role as a representation of continuum dynamics. The representation included drawings, but clearly those notations were not sufficient to support Maxwell's reasoning. The model includes notations along with their interpretations as hypothetical objects that can be considered, combined, and modified to have different properties and interactions. Study of the abstracted model, including exploration of variations of its basic structure, was possible and necessary for it to provide the basis of Maxwell's theoretical achievement. Models in which analogical thinking depends only on a hypothesized cognitive representation of a mapping or schema that connects two domains have not addressed the role of the representation as providing an object of analysis and hypothesis construction. We expect to learn more about these issues from a thorough comparison of the Maxwell example and others like it with models of analogical reasoning by schemata and structure-mapping. Our analysis should also provide a deeper understanding of the reciprocal processes through which formal representations and abstracted models are constructed in scientific discovery. We hope additionally that this exploration will provide helpful suggestions about ways in which abstracted models can be used more productively in the practice of science education.

ACKNOWLEDGEMENTS

Preparation of this paper was supported in part by National Science Foundation Scholars Award SES-8821422 to Nersessian and by National Science Foundation Grant BNS-8718918 to Greeno.

REFERENCES

Faraday, M. (1835-55). *Experimental Researches in Electricity*. Reprinted, New York: Dover.

Feynman, R. P., Leighton, R. B., & Sands, M. (1964). *The Feynman Lectures on Physics*. Reading, MA: Addison-Wesley.

Gentner, D. (1983). Structure-mapping: A theoretical framework. *Cognitive Science, 7*, 155-170.

Gick, M. L., & Holyoak, K, J. (1980). Analogical problem solving. *Cognitive Psychology, 12*, 306-355.

Gick, M. L., & Holyoak, K. J. (1983). Schema induction and analogical transfer. *Cognitive Psychology, 15*, 1-38.

Greeno, J. G. (1989). Situations, mental models, and generative knowledge. In D. Klahr, & K. Kotovsky (Eds.), *Complex information processing: The impact of Herbert A. Simon* (pp. 285-318). Hillsdale, N.J.: Lawrence Erlbaum Associates.

Jackson, J. D. (1962). *Classical Electrodynamics*. New York: Wiley.

Langley, P., Simon, H. A., Bradshaw, G. L., & Zitkow, J. M. (1987). *Scientific discovery: Computational explorations of the creative processes*. Cambridge, MA: MIT Press.

Maxwell, J. C. (1890). *The Scientific Papers of James Clerk Maxwell*, W. D. Niven, ed. Cambridge: Cambridge University.

Nersessian, N. J. (1984). *Faraday to Maxwell: Constructing Meaning in Scientific Theories*. Dordrecht: Kluwer.

Nersessian, N. J. (1988). Reasoning from imagery and analogy in scientific concept formation. In A. Fine & J. Leplin, eds. *PSA 1988*. East Lansing, MI: PSA.

Nersessian, N. J. (in press). *Methods of conceptual change in science: Imagistic and analogical reasoning*. Philosophica.

Panofsky, W. & Phillips, M. (1962). *Classical Electricity and Magnetism*. Reading, MA: Addison-Wesley.

Coulomb's Law:	$\text{div}\ \vec{D} = 4\pi\rho$
Ampère's Law:	$\vec{\text{curl}}\ \vec{H} = 4\pi\vec{J}$
Faraday's Law:	$\vec{\text{curl}}\ \vec{E} = -\partial\vec{B}/\partial t$
Absence of Free Magnetic Poles:	$\text{div}\ \vec{B} = 0$
Equation of Continuity:	$\text{div}\ \vec{J} + \partial\rho/\partial t = 0$

TABLE 1: Equations Maxwell began with according to standard textbook account

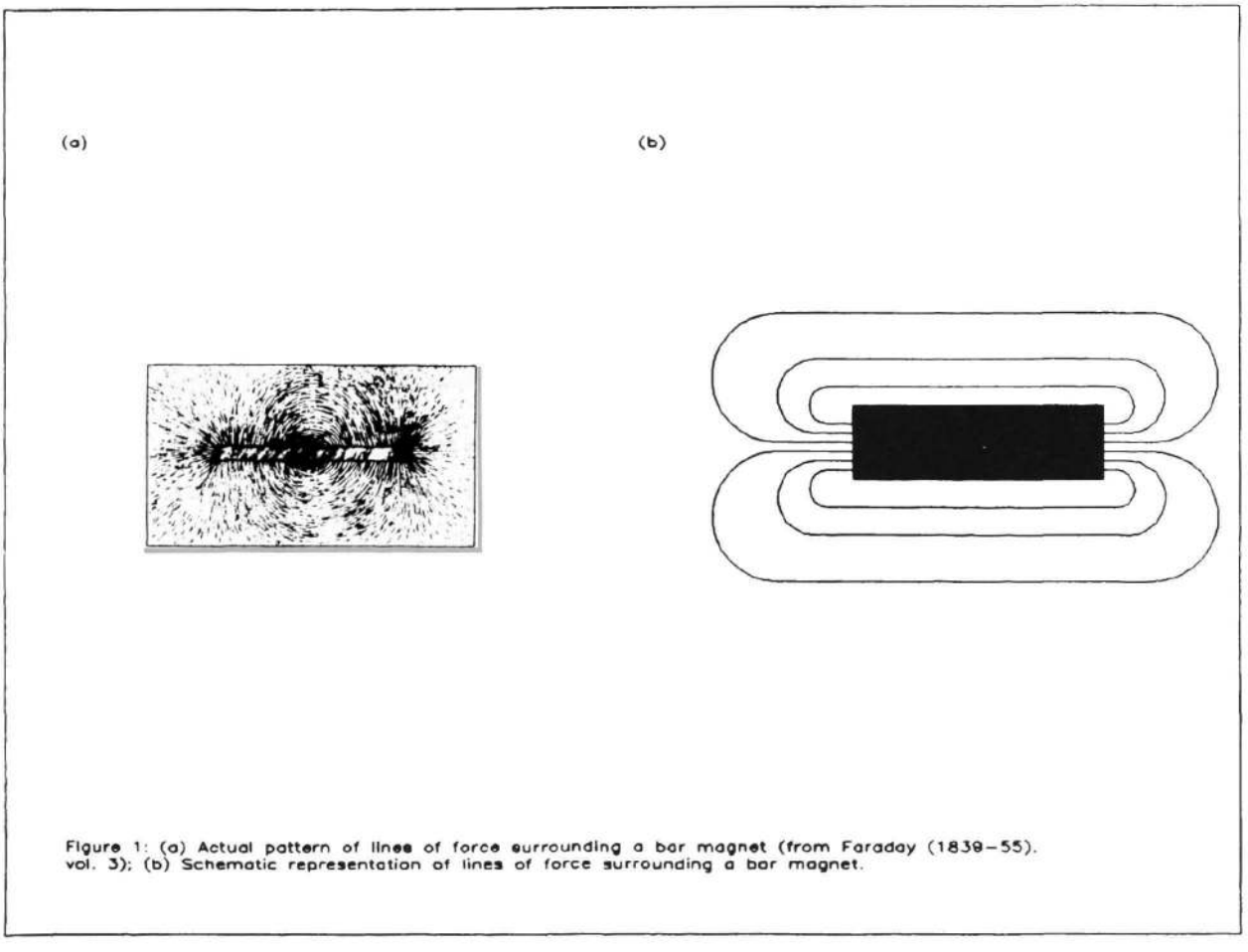


Figure 1: (a) Actual pattern of lines of force surrounding a bar magnet (from Faraday (1839–55), vol. 3); (b) Schematic representation of lines of force surrounding a bar magnet.

(a)

(b)

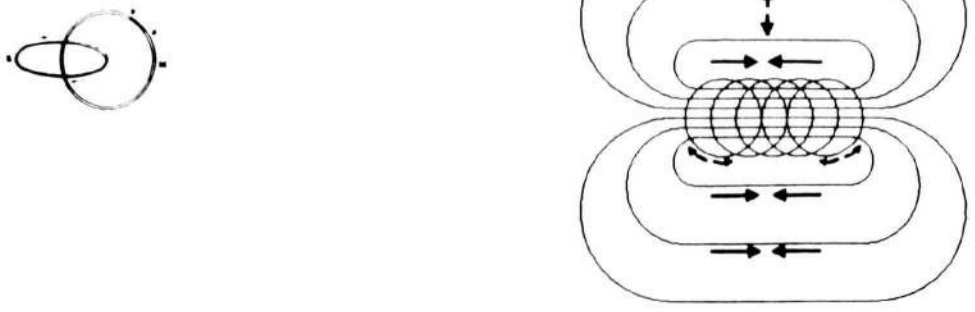


Figure 2: (a) Faraday's representation of the interconnectedness of electric currents and magnetic force (from Faraday (1839–55), vol. 3); (b) Schematic representation of the reciprocal relationship between magnetic lines of force and electric current lines.

(a)

(b)

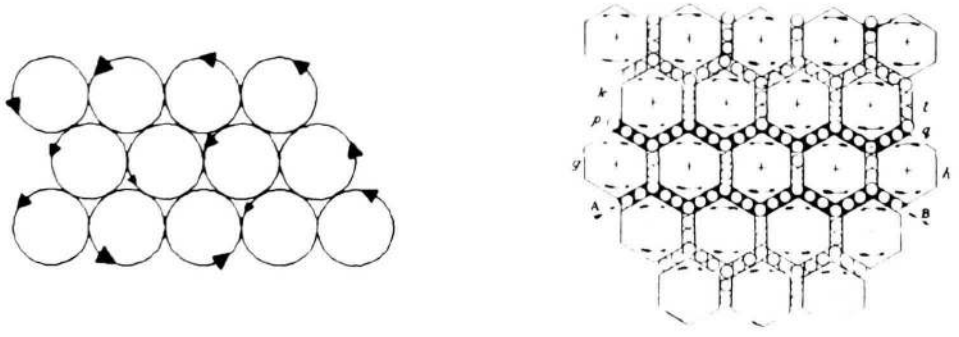


Figure 3: (a) Schematic representation of initial crude source retrieved by Maxwell; (b) Maxwell's representation of his fully elaborated "physical analogy" (from Maxwell (1881–2)).

Viewing Design as a Cooperative Task

Katia P. Sycara

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
katia@cs.cmu.edu

ABSTRACT

Design can be modeled as a multi-agent planning task where several agents that possess different expertise and evaluation criteria cooperate to produce a design. The differences may result in conflicts that have to be resolved during design. The process by which conflict resolution is achieved is negotiation. In this paper, we propose a model of group problem solving among cooperating experts that supports negotiation. The model incorporates accessing information from a case memory of existing designs, communication of design rationale, evaluation and critiquing of design decisions. Incremental design modifications are performed based on constraint relaxation and comparison of utilities.

INTRODUCTION

The design task can be described as taking a set of functional specifications and constraints, and producing an artifact representation whose behavior, when manufactured, conforms to the given specification description. Design constraints involve not only physical laws and domain principles that determine the behavior of the device but also restrictions and interactions arising from concerns such as cost, ease of manufacturing, ease of assembly and ease of maintaining the artifact. Taking these concerns into consideration in the initial design stages is known as concurrent engineering [DICE 89, Sriram 88][2]. Concurrent engineering is a team effort that requires the collaboration of teams of specialists representing relevant perspectives. Typically, each specialist has expertise in one area pertaining to the design, limited knowledge of the constraints and intentions of the other specialists and in general different evaluation criteria of the design. Hence, inconsistencies and conflicting views may occur. The specialists interact in a cooperative manner in order to effect tradeoffs and resolve inconsistencies. The final design is a compromise that has been agreed upon by all concerned agents[3]. Existing approaches to modeling concurrent design have primarily focused on investigating architectures for communication between various experts [DICE 89, Lander 88, Talukdar 88], or on conflict detection [Robinson 87, Sriram 88]. In this paper, we present a model of the group problem solving process of a team of concurrent engineering cooperating experts. The proposed team design model is inspired by our work in an adversarial domain, namely conflict resolution in labor management disputes [Sycara 87]. The model is currently being implemented in the CADET system that integrates Case-Based Reasoning, Qualitative Reasoning, and Constraint Propagation in the domain of mechanical design [Sycara 89a, Sycara 89b].

---

[1]This research has been supported by DARPA and AFOSR under contract number F49620-90-C-0003.

[2]It has been recently recognized that the practice of concurrent engineering is very advantageous in terms of reducing the time of a product from "concept to market". Industry is rapidly moving to adopt the paractice.

[3]We use the words "agent", "expert" and "specialist" interchangeably for the purposes of this paper.

The team design problem has the following characteristics:

- The global goal is to produce a design that is synthesized from contributions of different expertise, concerns and constraints
- During the design process, conflicts in the form of constraint violations could arise. If these conflicts are not resolved in a satisfactory manner, infeasible designs will occur.
- Disparate evaluations of (partial or complete) designs could surface as a result of different criteria used to evaluate designs from different perspectives. Typically, these criteria cannot be simultaneously and optimally satisfied. The design decisions that optimize one set of criteria could conflict with those that optimize another set. If these conflicts do not get resolved in a satisfactory fashion, design suboptimalities occur.
- The global goal is achieved by making the best tradeoffs on conflicting design goals and constraints.
- Because of the presence of conflicting constraints, goals and possibly evaluation criteria, it is impossible for each expert to optimize the overall design using only local information.
- Backtracking, resulting from infeasible designs, can be a major problem since it may result in invalidating design decisions that other agents have made.

As a result of the above characteristics, the final successful design can be viewed as a compromise[4]. that incorporates tradeoffs such as cost, ease of manufacturing and assembly, reliability and maintainability. The process through which design decisions are made by a team of experts is *negotiation.* Typically in manufacturing enterprises (e.g., [Bond 89]), after initial study of the design specifications, an initial design (called an initial cartoon) is created by the main designer. Subsequently, the specialists meet to negotiate proposed changes and tradeoffs with respect to the initial design. Compromises are suggested and discussed. The suggested compromises initiate further studies that necessitate additional meetings to reconcile continuing problem areas. This iterative process continues until all parties reach agreement[5]. Depending on particular decisions concerning tradeoffs, different designs will be produced. For example, the valve for a water tap could be a metallic threaded part or a plastic plug valve with a hole. There is a tradeoff between the low cost of the plastic valve and the high durability of the metal valve. Despite the difficulty of applying negotiation techniques, recorded conflicts and their resolution, namely previous similar design cases, provide a foundation for rationalizing designs.

Negotiation enters the design process at the following points:

- When the result of design decisions is an infeasible design (i.e. when constraint violations have been identified).
- When a design is feasible but suboptimal.
- When alternate approaches can achieve similar functional results.

---

[4]Pruitt [Pruitt 81] has identified two types of negotiation which are used by expert human negotiators to seek acceptable solutions and which may be applicable to machine agents: (a) compromise negotiation where each party makes concessions on its demands to facilitate agreement, and (b) integrative negotiation where the most important goals of each party are used to form innovative solutions, relinquishing, if necessary, secondary goals. In our view, both these negotiation types result in *compromise* solutions, in other words in partial goal satisfaction. Moreover, in typical negotiations a goal of secondary importance to one agent could be of primary importance to another because of different local evaluation functions. My use of the word "compromise" encompasses both these negotiation types.

[5]Throughout this process, time and cost determine the number of iterations allowed.

Negotiation is a process in which the parties iteratively exchange proposals and proposal justifications until an agreement is reached. During the negotiation process, the feasibility and desirability of proposed tradeoffs is evaluated and may result in incremental design adaptations. Thus, a negotiation model must be (a) iterative, (b) include mechanisms to incorporate feedback of the parties concerning evaluation of a proposal (partial design) from their point of view, (c) include criteria for judging whether progress in the negotiation is being made, and (d) incorporate negotiation protocols for the exchange of proposals, arguments and justifications of the proposed design decisions. The proposed model incorporates the above characteristics.

AN EXAMPLE OF EXPERT INTERACTION

Consider the process of designing a turbine blade. Some of the dominant specialties are aerodynamics, structural engineering, manufacturing and marketing. The blade design team operates within constraint ranges specified by the design team for the aircraft engine. The blade design team incorporates various concerns. The concern of aerodynamics is aerodynamic efficiency; for structural engineering it is reliability and safety; for manufacturing, it is ease and cost of manufacturing and testing; for marketing it is overall cost and customer satisfaction. The two variables of concern in a turbine blade that we consider are: (a) root radius, and (b) blade length. From the perspective of structural design, the bigger the root radius, the better since it decreases stress concentration. From the perspective of aerodynamics, the smaller the root radius, the better, since it increases aerodynamic efficiency. Concerning the length of the blade, from the point of view of structural design, the shorter the blade, the lower the tensile stresses; from the point of view of aerodynamics the longer the blade, the better the aerodynamics. On the other hand, if the blade is shorter, it makes for a lighter engine which is a desirable characteristic for aerodynamic efficiency. Thus, we see that the aerodynamics expert needs to make tradeoffs internal to its perspective. From the point of view of marketing, aerodynamic efficiency lowers the cost of operation of the aircraft, thus making it more attractive to customers. From the point of view of manufacturing, it is easier to manufacture shorter blades with bigger root radii.

The following is a simplified example dialogue of the various concerned perspectives in an attempt to arrive at a mutually satisfactory turbine blade design:

Aerodynamics (using case based reasoning) suggests particular values x and y for length and root radius of the blade. The suggested x and y values are within acceptable constraint ranges.

Structural engineering evaluates these values from its point of view and suggests values x' and y' where $x'<x$ and $y'>y$ (i.e., shortening the length and increasing the root radius) to increase safety.

Aerodynamics counters by saying that the values structural engineering suggested would considerably decrease aerodynamic efficiency.

Structural engineering counters that shorter blade makes engine lighter, thus also increasing efficiency.

Marketing says aerodynamic efficiency sells the product since it is less costly to operate.

Manufacturing supports structural by saying it is easier to manufacture short blades with big root radius.

Aerodynamics suggests that the materials engineering expert could try to investigate new materials that make the blade lighter, thus alleviating weight considerations.

Manufacturing says that new materials take lots of time to test and debug.

Structural engineering adds that new materials may introduce safety hazards that could go undetected.

This example illustrates the exchange of proposals for values of length and radius of the blade as well as the exchange of arguments and justifications in critiquing the proposed values. The nature of the resulting design will depend on (a) the ranges of various artifact constraints, (b) which constraints can be relaxed and in what ways, (c) the relative importance of various artifact-dependent goals (e.g., aerodynamic efficiency), (d) relative importance of various artifact-independent goals (e.g., safety), and (e) the way in which particular variable values contribute to the achievement of the goals.

THE MODEL

At the start of the team design process, an initial design is generated and presented to each expert, who evaluates it from its own point of view and registers its reactions (evaluations, objections and suggestions). At each negotiation iteration, the input is the set of conflicting concerns, violated constraints of the various design agents and the context of the design (e.g., constraints that have been handed to the design team from others). The final output is either a single agreed upon design or an indication of failure if the negotiating agents did not reach agreement within a particular number of iterations. The final output is reached through iterations of the following tasks: (a) proposal of an initial design, (b) arguments to support justification and critiquing of the design and (c) incremental modification and improvement. These tasks are performed using knowledge of existing designs and their characteristics, knowledge of physical laws and constraints, traces of design decisions made so far, and models of the expertise and concerns of the participating agents. The agents have access to a common case base of previous designs but because of their specialized knowledge each one accesses and evaluates a design from a particular *view*.

Negotiation, as seen in Figure 1, is performed through integration of Case-Based Reasoning (CBR) [Kolodner et al. 85, Sycara 87], use of multi-attribute utilities, called Preference Analysis, and constraint relaxation. These methods are employed in all negotiation tasks, namely in generation of an initial proposal, repair of a rejected proposal to formulate a counterproposal, and communication of justifications and objections. The process interleaves local computation and communication of computation results to other agents. One of the important concerns in distributed problem solving is minimization of communication overhead.

Use of previous design cases offers a reasoner (a) suggestions of how tradeoffs and resolutions have been made in the past, (b) failure avoidance advice (since failures are recorded in the design case), and (c) possible modifications and repairs to design suboptimalities. In addition, for group problem solving, having a memory of past problem solving experiences (successes and failures) has the following advantages [Sycara 89c]:

- Case-based inference minimizes the need for information exchange, thus minimizing communication overhead.
- Anticipating and avoiding problems through reasoning from past failures helps the agents minimize the exchange of proposals that will be rejected, thus minimizing backtracking.
- Reasoning from previous cases helps in recognizing problems or opportunities in a proposed design.
- If the repair of a past failure is also stored in memory, computation by each agent is

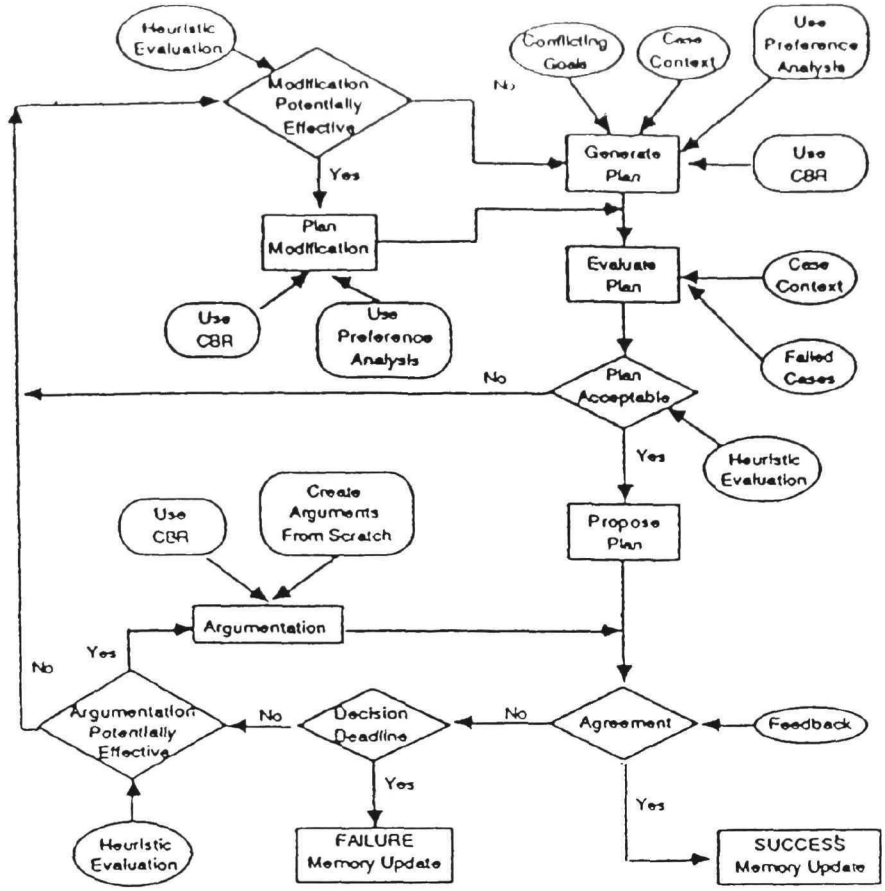


Figure 1: The Negotiation Process

minimized.

An integral part of the negotiation process is the ability of the agents to communicate their views and preferences as well as influence the decisions of other agents. Thus, a group problem solving model must have the ability to support (a) representing and maintaining belief models, (b) reasoning about agents' beliefs, and (c) influencing other agents' beliefs. The knowledge needed to perform these tasks is an agent's *belief* and *preference* structure. The belief structure of an agent, represented in a goal graph, consists of a collection of goals, goal importance and relationships among goals. The preference structure of an agent records its utilities associated with various potential decisions. The values in the goal graph are based on the constraints on the design and company policy. For example, after an airplane crash whose cause is traced to a defective engine, the importance of the safety goal increases for the marketing agent[6].

We represent an agent's belief structure as a directed acyclic graph where each node represents an agent's goal. Edges of the graph linking two goals represent the relationship between goals in terms of how one affects (positively or negatively) the achievement of the other. For example, aerodynamic efficiency positively affects lower operation costs. Associated with each node is:

[6]This has been observed in real situations, such as one of last year's airplane accidents involving an engine made by GE. The company gave increased important to testing procedures and design practices to increase engine reliability.

- a *sign* (+ or -) that denotes the desirability of an increase or decrease in that goal
- the *values* by which the attribute/goal should be increased or decreased
- the *importance* that the agent attaches to the goal
- the *feasibility* as perceived by the agent of achieving the goal

Directed edges connect subgoals to the higher level goals to which they contribute. A *contribution* value is associated with each directed edge denoting the contribution of the subgoal to the higher level goal. Contribution values range from -100% to +100%. A positive value means that the subgoal supports the achievement of the higher level goal by the denoted percentage. A negative contribution value has the interpretation that the subgoal is detrimental to the higher level goal. Sink nodes[7] are the highest level goals of an agent.

A path from node X to node Y in a goal graph constitutes a causal/justification chain that provides an explanation of the change in Y in terms of the change in X, assuming no other change has ocurred in the rest of the graph. For example, lengthening a turbine blade results in increasing aerodynamic efficiency, which in turn results in lower operating costs, thus resulting in increased marketability of the blade. By traversing goal graphs a reasoner can answer the following queries:

- Which goals are supported by a set of design decisions?
- Which design decisions are justified by a set of goals?

In addition to an agent's beliefs, the representation includes an estimate of its utilities for each attribute in the goal graph. Utilities express the *preference structure* of an agent [Sycara 88]. Moreover, utilities express the tradeoff structure among various attribute values associated with alternative designs. The (possibly nonlinear) utilities of individual attributes are combined to give an overall utility, the payoff, of an alternative. Being able to compare different alternatives enables a reasoner to choose the alternative that affords the maximum payoff. An integration algorithm traverses the belief structure to determine which way goal values should be moved to increase payoff and thus the acceptability of a resolution. Moreover, goal graph traversal allows an agent to discover alternative design decisions that support important goals thus leading to innovative designs.

THE NEGOTIATION PROTOCOL

The agents interact through message passing. The messages that the negotiating agents exchange contain the following information:

- The proposed design
- Justifications of design decisions
- Agreement or disagreement with the proposal
- Requests for additional information, such as with which issue in the proposed design the agent disagrees.
- Reasons for disagreement.
- Utilities/preferences of the agents associated with disagreed upon issues.

[7]Sink nodes have no out-going edges

We present in detail the communication protocol used in our system[8].

1. Agent1 communicates to agent2 a design proposal, as well as arguments and justifications in support of the proposal.

2. Agent2 uses the arguments and justifications communicated by agent1 to possibly modify its goal graph (e.g., change importance of goals, including possibly abandoning goals).

3. Agent2 evaluates the proposal from its point of view (using its constraints and utilities).

4. If the proposal satisfies agent2's local constraints and gives it payoff above a threshold, it communicates ACCEPT to agent1.

5. If not, agent2 generates a counterproposal by whatever problem solving means it has at its disposal (e.g., CBR, constraint relaxation).

6. Agent2 evaluates the counterproposal. If the counterproposal gives agent2 payoff above the threshold, agent2 communicates to agent1:

- The PORTION/ISSUES of the proposal that have been modified
- The REASON for modifying the previous proposal (e.g., value1 violates some of the agent2's hard constraints, a set of proposed values does not contribute enough to higher level goals of agent2).
- The COUNTERPROPOSAL and its PAYOFF.
- ARGUMENTS and JUSTIFICATIONS in favor of the counterproposal.

7. If the counterproposal does not give agent2 payoff above the threshold, agent2 goes to step 5.

8. If agent2 has exhausted all counterproposals it can generate through the methods of step 5, it traverses its goal graph to see whether there is another way to satisfy its higher level goals.

- If there is, it generates a counterproposal and goes to step 6.
- If there is not, it communicates FAILURE to agent1 (who now has to generate a modification and/or look for alternative ways in *its* goal graph).

CONCLUDING REMARKS

Design can be viewed as a multi-agent planning process involving multiple conjunctive and potentially conflicting goals. The agents have different expertise (e.g., mechanisms, hydraulics, assembly, testing) which results in viewing and evaluating designs using different, possibly conflicting criteria. The process of negotiation is used to propose and examine design decisions involving various tradeoffs. Negotiation is performed recursively at all stages of design and involves different design teams. We have proposed a model of group problem solving by cooperating experts that supports negotiation. The model is based on (a) knowledge of previous designs, (b) communication of design rationale, justifications and objections to proposed design decisions, (c) constraint propagation and relaxation, and (d) traversal of goal graphs.

---

[8]For simplicity, the protocol is presented for two agents, agent1, who initiates an initial design and agent2, who evaluates the design and possibly generates a counterproposal. The protocol generalizes to more than one agent that evaluates and suggests modifications.

REFERENCES

[Bond 89] Bond, A.H., and Ricci, R.J., "Cooperation in Aircraft Design," *Proceedings of the MIT/JSME Workshop on Cooperative Product Development*, MIT, Cambridge, Mass., November 1989.

[DICE 89] DICE: Initiative in Concurrent Engineering, "Red Book of Functional Specifications for the DICE Architecture," Tech. report, Concurrent Engineering Research Center, West Virginia University, February 1989.

[Kolodner et al. 85] Kolodner, J.L., Simpson, R.L., and Sycara-Cyranski, K., "A Process Model of Case-Based Reasoning in Problem Solving," *Proceedings of IJCAI-85*, Los Angeles, CA, 1985, pp. 284-290.

[Lander 88] Lander, S. and Lesser, V., "Negotiation to Resolve Conflicts Among Design Experts," *Proceedings of the AAAI-88 Workshop on AI in Design*, AAAI, St. Paul, MN., 1988.

[Pruitt 81] Pruitt, D. G., *Negotiation Behavior*, Academic Press, New York, N.Y., 1981.

[Robinson 87] Robinson, W., "Towards the formalization of specification design, Master's Thesis," Tech. report, University of Oregon, 1987.

[Sriram 88] Sriram, D., Logcher, R., and Groleau, N., "Cooperative Engineering Design," *Proceedings of the AAAI-88 Workshop on AI in Design*, AAAI, St. Paul, MN., 1988.

[Sycara 87] Sycara, K., *Resolving Adversarial Conflicts: An Approach Integrating Case-Based and Analytic Methods*, PhD dissertation, School of Information and Computer Science Georgia Institute of Technology, 1987.

[Sycara 88] Sycara, K., "Utility Theory in Conflict Resolution," *Annals of Operations Research*, Vol. 12, 1988, pp. 65-84.

[Sycara 89a] Sycara, K. and Navinchandra, D., "Integrating Case-Based Reasoning and Qualitative Reasoning in Engineering Design," in *Artificial Intelligence in Engineering Design*, J. Gero, ed., Computational Mechanics Publications, 1989.

[Sycara 89b] Sycara, K. and Navinchandra D., "A Process Model of Experience-Based Design," *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, MI., 1989.

[Sycara 89c] Sycara, K., "Multi-Agent Compromise via Negotiation," in *Distributed Artificial Intelligence, Volume II*, M. Huhns and L. Gasser, ed., Pittman Publishing Ltd and Morgan Kaufmann, 1989.

[Talukdar 88] Talukdar, S., Elfes, A., and Papanikolopoulos, N., "Concurrent design, simultaneous engineering and distributed problem solving," *Proceedings of the AAAI-88 Workshop on AI in Design*, AAAI, St. Paul, MN., 1988.

The Temporal nature of Scientific Discovery: The roles of Priming and Analogy

Kevin Dunbar & Christian D. Schunn
Department of Psychology, McGill University

Abstract: One of the most frequently mentioned sources of scientific hypotheses is analogy. Despite the attractiveness of this mechanism of discovery, there has been but a small success in demonstrating that people actually use analogies while solving problems. The study reported below attempted to foster analogical transfer in a scientific discovery task. Subjects worked on two problems that had the same type of underlying mechanism. On day 1, subjects discovered the mechanism that controls virus reproduction. On day 2, subjects returned to work on a problem in molecular genetics that had a similar underlying mechanism. The results showed that experience at discovering the virus mechanism did facilitate performance on the molecular genetics task. However, the verbal protocols do not indicate that subjects analogically mapped knowledge from the virus to the genetics domain. Rather, experience with the virus problem appeared to prime memory. It is argued that analogical mapping can be used flexibly in scientific discovery contexts and that primed knowledge structures can also provide access to relevant information when analogical mapping fails.

The origins of scientific hypotheses and theories have often been considered to be outside the purview of scientific investigation. Recently, however, a number of accounts of scientific discovery have appeared that provide detailed mechanisms of discovery (e.g., Holland, Holyoak, Nisbett and Thagard, 1986; Klahr & Dunbar, 1988; Langley, Bradshaw, Simon & Zytkow, 1987). One such account is the Scientific Discovery as Dual Search framework of Klahr and Dunbar (Klahr & Dunbar, 1988). Scientific Discovery as Dual Search (SDDS) is based upon the assumption that scientific reasoning consists of a search in two main problem spaces --an hypothesis and an experiment space. Using this framework, it has has been possible to explore the manner in which the current hypothesis guides experiment space search, and how experimental results in turn determine search of the hypothesis space. For example, Klahr & Dunbar (1988) have discovered that when subjects fail to find an hypothesis by searching the hypothesis space, they switch to searching the experiment space. Dunbar (1989) and Klahr, Dunbar, & Fay (1990) have discovered a number of heuristics for experiment space search.

While we are now acquiring a more detailed picture of the heuristics that guide experiment space search, relatively little is known about the heuristics governing the search of the hypothesis space. Klahr and Dunbar (1988) have shown that when subjects attempt to make a discovery, they initially search memory for an hypothesis. When an hypothesis has been found, the subjects then conduct experiments to see if the hypothesis is correct. If the hypothesis is incorrect, the subjects then either modify the hypothesis, search memory for a new hypothesis, or switch to a search of the experiment space. However the actual mechanisms underlying this search of the hypothesis space are unclear. The purpose of the research reported here was to provide a more detailed account of Hypothesis space search.

Research in other laboratories suggests two main mechanisms for hypothesis space search; analogical mapping (Gentner, 1983; Holland, Holyoak, Nisbett & Thagard 1986), and Remindings (Ross; 1989). Holland, et al. (1986) have emphasized the central role of

analogical mapping as a source of initial scientific hypotheses. Surprisingly, empirical research has demonstrated that subjects often fail to notice an analogical mapping when one is present. Most research that has demonstrated an inability of subjects to make use of analogies has been conducted using problems that demand little prior knowledge, and the subjects do not work on the problems for extensive amounts of time (e.g., Gick & Holyoak, 1983). When subjects are given more extensive training on a source domain there is a greater probability of analogical mapping occurring (e.g., Bassock & Holyoak, 1989). This suggests that the type of representation that subjects have of both the source and the target domain determines whether analogical mapping will occur. Gentner's (1983) structure mapping theory suggests that subjects will map from one domain to another only when they can abstract the relational structure of the source domain and the target domains. Both Gentner's theory and the goal oriented approach of Holyoak and Thagard (1989) suggest that subjects must have a detailed representation of the both the source and the target problems to make analogies.

Another mechanism that could be involved in mapping previously acquired knowledge on to a new problem is remindings; the use of an earlier example that the problem solver is reminded of while solving a problem. Remindings must be part of the analogical mapping process or could occur without being used in analogical mapping. Ross (1989) has proposed that when solving a problem there is a search of memory for related information. If the search is successful, the subject will be able to use the earlier example to solve the problem. If the earlier example has the same structural characteristics as the current problem the subject may be able to analogically map the earlier solution on to the current problem. If there is no structural overlap the subject could still use the reminding by fitting the previous solution to the current problem.

When applied to a scientific reasoning context, these views suggest that when a scientist has knowledge of a source domain, analogical mapping will occur only when a representation of the target domain has the same structure or the same subgoals as the source problem. When working on a problem, the scientist will initially retrieve some knowledge thought to be relevant to the problem. Experiments are then conducted. The experiments lead to a more detailed representation. When a more detailed representation of the problem is acquired, a search of memory for a structure or solution that will map onto the target domain will occur. While working on a problem a scientist may retrieve many different analogies to formulate hypotheses until a satisfactory solution to the problem is reached.

In sum, previous research suggests that if an attempt to map from the base to the target domain is made when subjects begin working on a problem, subjects will have superficial information about the target domain. Subjects should use their superficial knowledge of that domain to search memory for hypotheses relevant to the problem. As a result of experimentation, subjects should learn that their initial hypotheses do not hold. Then, as they learn more about the relations among the elements of the target problem --through experimentation-- they should develop a more detailed representation of the problem. This should make it possible to search memory for a structurally similar solution, or solutions that have been used to solve similar problems.

The Molecular Biology domain

To investigate the temporal course of the development of hypotheses a task that fosters the development of changing representations of a problem is needed. Rather than invent an arbitrary problem, two problems from molecular biology that involve similar underlying mechanisms were used. The source problem was one of discovering why viruses are sometimes dormant (do not reproduce), and other times are active (reproduce rapidly). Molecular biologists have discovered that dormant viruses secrete enzymes that inhibit virus reproduction. This is a form of negative regulation. The target problem also involved

negative regulation. The target problem was to discover how genes are controlled by other genes. The specific question addressed was why genes secrete beta-galactosidase only when there is lactose present. Again the mechanism underlying this is negative regulation (Jacob & Monod, 1961). Monod and Jacob were awarded the Nobel prize for discovering that the control mechanism in viruses and in lactose is negative regulation.

To investigate the time course of analogical mapping a paradigm developed by Dunbar (1989) was used. In the Dunbar (1989) study subjects were taught some elementary knowledge about molecular genetics. Then, subjects were taught how to conduct molecular genetics experiments on the computer. Then they were given the problem of discovering the way that genes control other genes. Subjects tended to use their knowledge of the genetics domain to formulate their initial hypotheses. Subsequent hypotheses were formed by inducing the concept of negative regulation from patterns of data or memory search.

In the current study, subjects were taught about viruses and genetics and were shown how to conduct simulated experiments on the computer. For both the virus and the genetics problem subjects were asked to discover the control mechanism. The tasks that the subjects were given and the potential analogical mappings will now be described in detail.

The Experiment

Subjects. Sixty McGill undergraduates were paid to participate in the experiment. All subjects had taken an introductory biology course and they were not familiar with gene regulation in bacteria or viruses. Their knowledge of molecular genetics consisted of knowing that DNA and RNA exist.

Procedure. There were three conditions. In the *No-Virus* condition, subjects participated only in the gene regulation problem. There were two experimental conditions --*Negative-Virus* and *Correlated-Virus*. In these conditions subjects attempted to discover why viruses are sometimes active and sometimes dormant. The virus problem was presented on day 1, and the genetics problem was presented on day 2. Subjects conducted their experiments on a MacIntosh II computer. The display was highly interactive: subjects conducted experiments by using menus and selecting various options. Once the experiment was designed subjects could conduct the experiment and monitor the results. A permanent record of all experiments and results was available after each experiment was conducted.

Day 1. Virus Problem. The virus problem was carried out in three phases. First, the subjects were taught some basic facts about viruses and biochemistry, and were shown two methods of determining whether a chemical is involved in virus reproduction. One method was to view the amount of a chemical that is present in an active or dormant virus. By doing this the subject could see whether there was a relationship between the amount of a chemical and the number of viruses present. The second method was to add a chemical to an active or dormant virus and monitor its effect on the number of viruses present.

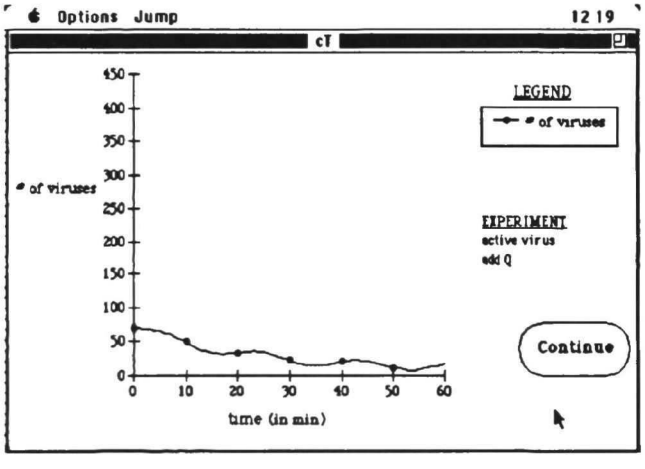
Second, the subjects were instructed on how to give a verbal protocol. Third, the virus problem was given to the subjects. They were told that viruses were sometimes active and sometimes dormant, and that biochemists thought that one of three chemicals that the virus can secrete was responsible for making the virus active or dormant. Subjects were asked to discover which of the three chemicals causes the virus to be active or dormant and what the control mechanism was.

For the *Negative-Virus* group the mechanism causing the virus to become dormant was negative regulation: The dormant viruses secrete a chemical that prevents generation of new viruses. For the *Correlated-Virus* group, there was a positive correlation between the amount of Q that was present and the number of viruses. In this case, Q was a by-product of the number of viruses present. There was no causal mechanism for this group; this was a control condition. Subjects worked on this problem until they felt that they had

discovered the correct answer. Subjects were not told if they had discovered the correct mechanism. All subjects were asked to return for another problem the next day.

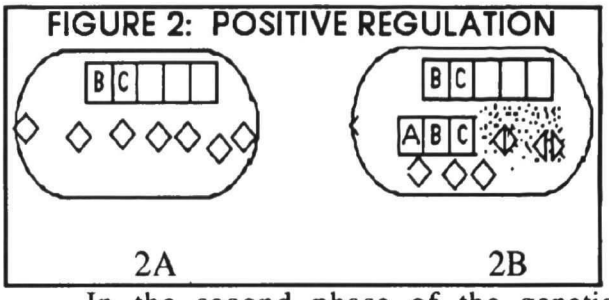
There were 12 possible experiments that could be conducted. in the virus problem. The small number of experiments made it extremely likely that subjects would conduct all possible experiments. In particular, in the *Negative-Virus* condition, subjects should conduct an experiment where Q is added to an active virus and the number of viruses decreases, (see Figure 1). This would suggest that Q inhibits virus reproduction. Due to the small size of the experiment space it was expected that most subjects would discover the correct mechanism.

FIGURE 1: NEGATIVE-VIRUS CONDITION



Day 2. Genetics problem. All three groups of subjects were given the genetics problem. There were three phases in the genetics problem. First, the subjects were taught some basic facts about molecular biology. They were told that certain genes control the activities of other genes by switching them on when there is a nutrient present. This is an example of positive regulation --the regulator gene senses that there is a nutrient present and then releases substances that instruct other genes to secrete enzymes that can utilize the nutrient. The example of positive regulation given to the subjects can be seen in Figure 2. Here, the A gene switches on the enzyme producing genes. When the A gene is absent and fructose is present no enzyme is produced (FIG 2A). When the A gene is present in a diploid cell the cell secretes an enzyme that breaks down the fructose (FIG 2B).

FIGURE 2: POSITIVE REGULATION



In the second phase of the genetics problem, subjects were shown how to give a verbal protocol. In the third phase, subjects were presented with two findings about a set of genes labelled I, O, and P: The enzyme producing genes secrete an enzyme when there is lactose present, and do not secrete the enzyme when there is no lactose. Subjects were asked to discover how the enzyme producing genes are controlled. The presentation of the I,P,O genes was identical to the A,B,C genes. The differences were (i) that there were different letters on the genes (ii) the nutrient was lactose rather than fructose, and (iii) the enzyme secreted was betagalactosidase rather than delta. Finally, unbeknownst to the subject, there was a different underlying mechanism --negative regulation rather than positive regulation. The subjects have to discover that the I and O genes negatively regulate (i.e., inhibit) the activity of the enzyme producing genes until a nutrient is present. Subjects must also learn that the I gene can be present on either the male or female chromosome, but that the O gene can only be on the O gene.

There are 120 possible experiments that can be conducted (6 amounts of nutrient x 20 genetic combinations). For a normal Ecoli, the amount of enzyme produced is half whatever the amount of nutrient is. The I gene mutants (designated I- mutants) produce an output of 876, and O- mutants produce an output of 527. The I- and O- mutants produce this amount of enzyme regardless of the amount of nutrient administered. Thus, I- and O- nutrients will produce enzymes even when

no nutrient is administered. This is a strong clue for negative regulation being involved. In this study, the P gene plays no role at all.

Subjects were expected to make a number of different mappings from both the virus and the genetics domains during the course of the experiment. First, there should be a mapping from the positive regulation example given at the start of the genetics problem. The positive regulation example had a virtually identical structure to the target problem. There were three regulator genes, three enzyme producing genes, nutrients with similar names (lactose and fructose), and enzyme products that also have similar names (beta and delta). Thus, as in the Dunbar (1989) study, it was expected that initial hypotheses would be of positive regulation.

Once subjects discover that none of the genes work by positive regulation there should be a search of the hypothesis space for another mechanism. For the *Negative-Virus* group, it was expected that the presence of large amounts of enzyme when either the I or O genes were mutants would be equivalent to the situation of large numbers of virus being present when Q was absent. At this point, subjects should mention that the genetics problem is like the virus problem. The subjects should then map over the concept of negative regulation from the virus to the genetics domain.

If the virus problem is used to help solve the genetics problem, then it could occur in a number of stages. First, subjects would have to extract a representation such as *X inhibits Y* from *Q inhibits virus production.* Then they should apply this relation to the genetics problem. Second, even with this mapping, the problem is not solved; the *X inhibits Y* relation must be modified to include two genes (I and O), and the that the I and O genes inhibit in different ways. Thus, subjects should retrieve the *X inhibits Y* relation, and then modify it (by experimentation) to fit the current problem. The genetics problem not only involves a memory search for hypotheses, but also a modification of hypotheses to solve the problem.

Results

Day 1 (virus problem). All of the 20 subjects in the *Correlated-Virus* condition discovered that the positive non-causal relationship between the amount of Q and the number of viruses. Sixteen of the 20 subjects in the *Negative-Virus* group discovered that the dormant virus secretes a chemical that inhibits virus reproduction. The four subjects that did not discover negative regulation concluded that there was a complex interaction of chemicals that switch on reproduction.

Day 2 (genetics problem). As shown in Table 1, for the *No-virus* group, 7 out of 20 subjects discovered that the I and O genes inhibit and that the P gene plays no role. For the *correlated-virus* condition, the results were almost identical: 7 subjects discovered the roles of I, O, and P. This indicates that the experience with conducting virus experiments on day 1 had no beneficial effect on performance. Results for the *Negative-Virus* group were very different; 12 of the subjects discovered the mechanism of genetic control. This indicates that discovery of the negative regulation of viruses on day 1 did indeed benefit performance on the genetics problem.

Table 1
Subjects discovering genetics mechanism

Condition		
	No-Virus	7 (35%)
	Correlated virus	7 (35%)
	Negative virus	12 (60%)

As can be seen from Table 2, 12 of the subjects who discovered negative regulation in the virus problem also discovered negative regulation on the genetics problem. Four of the subjects who discovered negative regulation in the virus problem did not discover negative regulation in the genetics problem. These subjects did mention inhibition, but also proposed a positive regulation mechanism.

Table 2
Effects of discovering virus mechanism on discovering genetics mechanism

	discover genetics	miss genetics
discover virus	12	4
miss virus	0	4

Initial hypotheses. It is possible to determine whether subjects immediately saw if the mechanism that they learned in the virus problem was applicable to the genetics problem by examining initial hypotheses. None of the subjects in the *No-Virus* or the *Correlated-Virus* conditions initially proposed negative regulation as a potential mechanism. Only one of the subjects in the *Negative-Virus* condition proposed that negative regulation as an initial hypothesis. This indicates that subjects initial mappings were not from the virus domain. Instead, initial hypotheses were taken from the example demonstrating positive regulation at the beginning of the genetics problem.

Subsequent hypotheses. As mentioned above, previous research using this paradigm (Dunbar, 1989) has shown that subjects discover negative regulation using two main strategies. One is by searching memory for an appropriate mechanism and the other is by inducing the mechanism from a series of experimental results. If subjects were using the knowledge obtained from the virus experiment, then subjects should discover the concept of negative regulation by a memory search, rather than by inducing negative regulation from a series of experimental results

We were able to categorize performance as memory search by one of two criteria. One criterion was whether subjects stated that they were searching memory for hypotheses or that the genetics problem reminded them of the virus problem. None of the subjects in either the *Correlated-Virus* or the *Negative-Virus* groups mentioned the virus experiment while they were working on the genetics problem. The other criterion was if on the first experiment that was inconsistent with positive regulation, subjects proposed negative regulation. For example, when an I-experiment is conducted there is a very large output of betagalactosidase. This result is inconsistent with positive regulation. All three groups of subjects conducted an I- or an O- experiment as one of their first four experiments. In the *No-virus* condition, 4 subjects suggested negative regulation after their first I-experiment. In the *Correlated Virus* condition 4 subjects proposed negative regulation. By contrast, 11 subjects in the *Negative-virus* condition proposed negative regulation .

Incorrect final hypotheses. It is also possible that the virus problem influenced the types of incorrect hypotheses proposed. In the *No-Virus* condition only 5 of the 13 subjects mentioned negative regulation. In the *Correlated-Virus* condition 4 of the 13 subjects mentioned negative regulation. In the *Negative-Virus* condition 8 subjects failed to propose the correct mechanism. The 4 subjects who missed both the virus and the genetics problem never mentioned inhibition. Three of the 4 subjects who solved the virus problem but did not solve the genetics problem mentioned negative regulation. Thus, all but one of the subjects who solved the virus problem proposed some form of negative regulation in the genetics problem.

One possible explanation for the better performance of the *Negative-Virus* group could be that these subjects learned various experimental skills while working on the virus problem. The performance of the *Correlated-Virus* group does not support this explanation. If subjects acquired knowledge of how to conduct experiments during the virus experiment, then both the *Correlated-Virus* and *Negative-virus* groups should have benefitted. As only the *Negative-virus* group showed improved performance, this explanation seems unlikely.

Discussion

The results of this study illustrate the dynamic nature of the scientific discovery process. Initial hypotheses were formed by analogically mapping from the same domain. When initial hypotheses were disconfirmed there was a search of memory for an hypothesis that could account for the discrepant results. This search was not based on superficial features of the problem. Subjects searched for a mechanism that could account for the current results. Once the mechanism of negative regulation had been retrieved it had to be modified to fit the current context.

Subjects in the *Negative-Virus* condition did not appear to discover negative regulation by analogical mapping from the virus domain. If subjects were mapping knowledge over from the virus domain, then there should have been a reference in the verbal protocols to the virus domain prior to proposing an inhibitory mechanism. None of the subjects made any reference to the virus domain. Furthermore, at the end of the experiment all subjects were asked whether they had thought of the virus experiment while working on the genetics problem. All the subjects claimed that they had not thought of the virus experiment. When asked if they thought that the virus problem had helped them on the genetics problem almost all subjects stated that the experience with the genetics problem helped them design better experiments. However, the fact that the performance of the subjects *No-Virus* condition was virtually identical to that of the subjects in the *Correlated-Virus* condition indicates that experience with the virus problem *per se.* had no effect on experimental strategies. In sum, these results appear to indicate that there is no direct mapping from the virus to the genetics problem.

The fact that subjects made no explicit mapping from the virus to the genetics problem was surprising. Most theories of analogical reasoning assume that when mapping occurs, it is a conscious activity that occurs by searching memory for a match between the target problem and other possibly relevant mappings. The results of this study suggest a different mechanism: The virus problem may have primed the concept of negative regulation. When the subjects discovered that the mechanism was not one of positive regulation, they engaged in a memory search. The memory search finds negative regulation because it has been primed by the virus problem. This priming mechanism would account for the fact that (a) subjects initial hypotheses were not concerned with negative regulation, and (b) no evidence of an explicit mapping from the virus to the genetics domain appeared in the protocols.

A number of mechanisms could have produced the results obtained in this study. One is priming of a pre-existing knowledge structure. Cheng and Holyoak (1986), and Fong and Nisbett (1989) have proposed that there are a number of well ingrained concepts in memory that they have labelled 'pragmatic reasoning schemas.' A concept such as negative regulation could be one of these types of schemas. In the current study, solving the virus problem may have primed the concept of negative regulation, making it more likely to be retrieved while solving the genetics problem.

If the prior learning episode primes a concept that is already present in memory, then it suggests that this knowledge structure priming will occur only for concepts that are represented in memory prior to participation in the experiment. However, studies of implicit learning (cf. Reber, 1989) have discovered that when subjects observe a number of instances that follow a rule, they form an abstract representation of the rule that is not tied to the particular context under which the 'rule' was learned. A similar type of learning may have occurred in this study. By learning through experimentation, subjects may have acquired an abstract concept that is missing the contextual information about how the concept was acquired. When working on the genetics problem this abstract representation may have been accessed and used to solve the problem. As the concept had no contextual information associated with it, no remindings occurred. We are currently investigating this hypothesis..

Finally, we can now provide a more dynamic account of the heuristics governing hypothesis space search in SDDS. First there is a search for an hypothesis that can account for an instance that has similar features to the target problem. These initial hypotheses can be based upon the superficial features of the target problem and also upon what little is known about the deep structure features of the problem (cf. Faries & Reiser, 1988). Second, if this search is successful, then the hypothesis is tested. Third, if the hypothesis fails the test, then there is a search for other potential hypotheses, if none are found there is an attempt to modify the initial hypothesis to accommodate the data. This later search is

qualatitively different from the first search; subjects now have more knowledge of the target problem and can search for an underlying mechanism rather than a match for superficial features. When a mechanism is being searched for, the current state of activation of all mechanisms in memory will determine which mechanisms will be retrieved. If, as in the current study, a mechanism has been primed by a prior problem, then it will be retrieved.

References

Bassock, M., & Holyoak, K.J. (1989).Interdomain transfer between isomorphic topics in algebra and physics. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15*, 153-1666.

Cheng, P, & Holyoak, K. (1985). Pragmatic reasoning schemas. *Cognitive Psychology,* 17, 391-416.

Dunbar, K. (1989). Scientific reasoning strategies in a simulated Molecular genetics environment. I*n the proceedings of the 11th annual meeting of the Cognitive Science society, MI Ann Arbor*, 426-433.

Dunbar, K., & Klahr, D. (1989). Developmental differences in Scientific Discovery Strategies. In D. Klahr, & K. Kotovsky (Eds.). *Simon and Cognition: Proceedings of the 21st Carnegie-Mellon Symposium on Cognition.* Erlbaum: Hillsdale, New Jersey.

Faries, J.M., & Reiser, B.J. (1988) Access and use of previous solutions in a problem solving situation. *In the proceedings of the tenth annual meeting of the Cognitive Science Society. Montreal, Quebec.*

Fong, G.T., & Nisbett (1990). Immediate and delayed transfer of training effects in statistical reasoning. Manuscript submitted for publication.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7 , 155-170.

Gentner, D. (1989). The mechanisms of analogical learning. In S. Vosniadou, & A. Ortony (Eds.), *Similarity and Analogical reasoning*. Cambridge, Cambridge University Press.

Gick, M.L., & Holyoak, K.J. (1983). Schema induction and analogical transfer. *Cognitive Psychology, 15,* 1-38.

Holland, J., Holyoak, K., Nisbett, R.E., & Thagard, P. (1986). *Induction: Processes of Inference, Learning, and Discovery* . Cambridge, MA: MIT Press.

Holyoak, K. J. & Thagard, P.J. (1989). A computational model of analogical problem solving. In S. Vosniadou, & A. Ortony (Eds.), *Similarity and Analogical reasoning.* Cambridge, Cambridge University Press.

Jacob, F., & Monod, J. (1961). Genetic regulatory mechanisms in the synthesis of proteins. *Journal of Molecular Biology*, 3, 318-356.

Klahr, D., & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Science* . 12, 1-48.

Klahr, D., Dunbar, K, & Fay. (1990). Designing good experiments to test "bad" hypotheses. In J. Shrager, & P. Langley (Eds.), *Computational models of discovery and theory formation.* Ann Arbor, MI: Morgan Kaufmann.

Langley, P, Simon, H.A., Bradshaw, G.L., & Zytkow, J. (1987). *Scientific Discovery: Computational explorations of the creative processes.* Cambridge, MA: MIT press.

Reber, A. S. (1989) Implicit learning and tacit knowledge. *Journal of Experimental Psychology: General, 118,* 219-235.

Ross, B.H. (1989). Remindings in learning and instruction. In S. Vosniadou, & A. Ortony (Eds.), *Similarity and Analogical reasoning.* Cambridge, Cambridge University Press.

Notes

This research was supported by grant number OGP0037356 from the National Sciences and Engineering Council Canada to Kevin Dunbar and a summer NSERC research award to Christian D. Schunn.

Correspondence can be addressed to:Kevin Dunbar, Department of Psychology, McGill University, Montreal, Quebec, Canada H3A 1B1
Email: DUNBAR@HEBB.PSYCH.MCGILL.CA

Reasoning directly from cases in a case-based planner

Robert McCartney
Department of Computer Science and Engineering
University of Connecticut, U-155
Storrs, CT 06269-3155
robert@uconn.edu *

Abstract

A good deal of the reasoning done in a case-based planning system can be done directly from (episodic) cases, as opposed to specialized memory structures. In this paper, we examine the issues involved in such direct reasoning including how this representation can support multiple uses, and what role execution plays in such a framework. We illustrate our points using COOKIE, a direct case-based planner in the food preparation domain.

1 Introduction

In this paper, we examine the issues regarding the direct use of cases by a case-based planner. By direct use, we mean performing the underlying reasoning in a system by manipulating representations of episodes in memory, rather than an intermediate description or specialized structure. While these issues are shared by all case-based reasoners, we focus particularly on case-based planning. We examine case representation, how cases are used for different purposes, and the role of execution; these examinations are based on work with COOKIE, a case-based system that plans and monitors execution in the domain of meal planning and preparation.

What is case-based reasoning?

Case-based reasoning is the solving of problems through the reuse of experience—when faced with a problem situation, the problem solver retrieves a similar situation (with its solution) from memory, then adapts the previous solution to solve the current problem [Riesbeck and Schank, 1989]. This can be distinguished (in theory) from rule-based reasoning, since we reason from cases corresponding to real episodes rather than from rules which are distillations of experience. In fact, this separation is muddied in practice, as case-based systems include cases that are abstractions based on a number of real episodes, and rule-based systems include rules that are in fact grounded in a single experience. Case-based reasoning has been used in a variety of domains for a variety of tasks; it has showed the most promise in situations characterized by uncertainty, lack of a complete domain theory, and/or computational constraints, where more traditional approaches have had little success.

*This research has been supported by Booth Research Center grant BG-6. The author gratefully acknowledges the assistance of Kate Sanders, Mallory Selfridge, and Karl Wurst.

Desiderata for a case-based planner

Case-based planning (CBP) is an attempt to solve planning problems by reusing previous episodes. The planning problem is to find a sequence of primitive actions that leads to some specified results—goals to be accomplished and constraints to be satisfied. A case-based planner does this by remembering experiences. At essence, a case-based planner follows the principles given by Hammond [Hammond, 1988]:

> *If it worked, use it again,* and
>
> *If it didn't work, remember not to do it again.*

To these, we add a principle of our own:

> *If my plan fails, I should figure out why.*

These principles can be translated into the three basic functions that a case-based planning system should be able to accomplish.

First, a CBP system should be able to generate plans given it has succeeded in a similar situation. This process is one of retrieving the similar situation (case), then performing transformations on the case until it matches the description of the current problem. Once it matches, "use it again" becomes a simple reapplication of the sequence of steps. For this to be effective, the transformations applied should be equivalence preserving, at least in regard to the goals and constraints.

Second, a CBP system should be able to recognize that a particular approach failed before, so should not be used again. Simply not retrieving failures for adaptation is insufficient here—if we transform a (successful) plan into one that fails, we would like to avoid doing it in the future.

The third principle relates to the second—to avoid failure in the future, we need to understand why we failed. If we can anticipate failure, then we can either use a different plan or modify our plan to avoid the problem. The principle as given only works in a negative way, but sometimes plans have unexpected good results. A more general principle can be based on expectation failures [Schank, 1982] rather than plan failures:

> If something odd happens, figure out why and remember it.

Overview of COOKIE

COOKIE is a case-based system that plans and monitors the preparation of food. Previous work (notably CHEF [Hammond, 1986] and JULIA [Kolodner, 1987]) has demonstrated the benefits of doing planning research in this domain, which is both rich and unpredictable. COOKIE's input is a set of goals and constraints: it first produces a plan to satisfy the input, then monitors the execution of that plan (performing execution-time repairs of the plan, if necessary), then incorporates the results of that execution into its memory. Execution is done externally to the system by human cooks using real food; case-memory is comprised of these episodes and other real cookings taken from transcripts. There is no causal reasoner for analyzing failure, nor is there a simulator that can be used to execute cookings. Explanations of anomalous behavior are built from the assumptions used to predict the non-anomalous behavior and from adaptations of other explanations found in cooking transcripts. COOKIE is designed to reasoning directly from its cases as much as possible, and relies on few other mechanisms. Specific examples of COOKIE's behavior relating to this paper's topic are given in the next few sections.

The rest of this paper is arranged (in order) around the following questions:

- What is an episodic case representation?

- How can an episodic representation scheme support multiple uses in a planner (generation, projection, recognition, explanation, and failure recovery)?
- What is the role of execution in a direct case-based planner?

Finally, in the conclusions, we discuss possible roles for abstraction in a case-based planner.

2 Case representation

Episodic representation—what it is, what it involves

Informally, an episodic representation is one that allows reconstruction of an episode as a story—what happened, when the various things happened relative to each other, what the results were, and so forth. More concretely, an episodic representation is a reasonably complete statement of the facts that is neither simplified nor abstracted. For any domain, the representation includes all of the facts about an episode that are likely to be useful in any future reasoning task. In the food preparation domain, this includes all of the cook-food interactions, as well as seemingly unrelated cook actions that occur during the cooking. For example, if a cook answers the telephone during meal preparation and talks for five minutes to "Dialing for Dollars", that fact is part of the episode, not just that the cook was out of the kitchen for five minutes.

Reasoning directly from cases

In a case-based reasoning system (or more generally, any reasoning system where the rules are ultimately grounded in episodes), we have two options: reasoning by direct manipulation of episodes, or reasoning from abstractions and/or simplifications of episodes. There are advantages and disadvantages to each approach. On the one hand, an episodic representation is likely to contain a good deal of irrelevant information for a particular task; if we want to explain burned biscuits, the cook not monitoring the food for five minutes is enough information, and the facts that it was a phone call, the exact time, that it was from "Dialing for Dollars" and so forth are unnecessary and could have been simplified out of the case. On the other hand, a simplification or abstraction may not have retained the right information. Suppose the above dinner turns out to be poisoned, killing the cook, and his beneficiary happens to work for the television station broadcasting "Dialing for Dollars"—the simplification to "the cook not monitoring the food for five minutes" precludes the obvious explanation of said beneficiary arranging for the phone call so he could poison the meal while the cook was distracted. *The main advantage in reasoning from episodes is that the information used can be extracted when the reasoning task and its context are established, rather than at the time the episode is stored.* This does not rule out a memory where episodes are stored at multiple levels of abstraction, precomputing those that are likely to be useful, but unless we can be certain of precomputing all useful abstractions, the need for direct reasoning cannot be eliminated.

Representation scheme in COOKIE

The representation language used in COOKIE is a temporal propositional logic. For example, (occur ev3 (do chef1 (stir soup5 saucepan1 spoon23)) t1 t2) means that event ev3, chef1 stirring soup5 in saucepan1 with spoon23, occurred beginning at time t1 and ending at time t2. Cases describing cooking episodes are propositional, and contain goals, descriptions of input, output, and (possibly) intermediate states, actions, a critique, and (if applicable) problems encountered and their associated repairs. An example, broiled steak with fried onions, is given in Figure 1. Propositions that are true at the same (specific) times are grouped together here. This is purely a syntactic aid to

```
(cooking steak-and-onions1
  (initial (raw steak23)(raw onion24)(cold frypan1) (quantity steak23 1lb)
           (thickness steak23 1.5 in) (quantity onion24 1 medium) (quantity oil25 1 Tb)
  (goals  (cooked steak23) (cooked onion24) (shape onion24 rings)))
  (final  (cooked steak23) (cooked onion24) (shape onion24 rings)
           (rating steak-and-onions1 success)))
  (facts  (inst steak23 beefsteak) (inst onion24 onion) (inst oil25 corn-oil)
  (events (occur ev0 (do chef1 (slice onion24 thin knife2)) 0 1)
           (occur ev1 (do chef1 (heat burner2 high)) 4 8)
           (occur ev2 (do chef1 (heat broiler1 high)) 0 8)
           (occur ev3 (do chef1 (put oil25 frypan1)) 4 4)
           (occur ev4 (do chef1 (put frypan1 burner2)) 4 4)
           (occur ev5 (do chef1 (put steak23 b-pan1)) 1 1)
           (occur ev6 (do chef1 (put b-pan1 broiler1)) 1 1)
           (occur ev7 (do chef1 (flip steak23 b-pan1 fork8)) 5 5)
           (occur ev8 (do chef1 (put onion24 frypan1)) 5 5)
           (occur ev9 (do chef1 (occasional (stir onion24 frypan1 fork8))) 5 8)
           (occur ev10 (do chef1 (remove onion24 frypan1)) 8 8)
           (occur ev11 (do chef1 (remove steak23 b-pan1)) 8 8)))
```

Figure 1: Broiled steak and onion rings.

the user; internally, a case is a conjunction of propositions with individual temporal characteristics. This case is a relatively simple one; other cases include (among other things) termination tests for events, underlying assumptions used in planning the episode, and expectation failures with their associated real-time repairs.

3 Multiple purposes from a single representation scheme

One benefit claimed for an episodic representation is that it is *task-neutral*: the same representation should be useful for a variety of reasoning tasks [McCartney and Sanders, 1990]. In COOKIE, the episodic representation directly supports plan generation and projection, explanation, and failure recovery during execution, and interacts with low-level abstractions during plan recognition.

Plan generation and projection

Plan generation (coming up with a sequence of actions to perform a task) and projection (predicting the effects of a proposed set of actions) in COOKIE are done by performing transformations on cooking episodes. The transformed episode corresponds to the generated plan (actions and expectations) and is processed directly by the execution module (see Section 4).

Projection is based on the assumption that the objects in the transformed episode will behave as their counterparts did in the original episode, so any observed facts in the original episode become expected facts (that is, the projections) in the transformed episode. For a given situation, there is a set of permissible transformations whose application should lead to behavioral equivalence in the transformed case. Suppose, for example, we want a plan for broiled hamburger and onions, and can use the following transformations:

(x→y: if (and (inst x z) (inst y z)(food z)))

(x→y: if (and (inst y ground-beef)
(inst x beefsteak)
(shape y patty)))

These correspond to the assumptions that food is fungible and we can substitute hamburger for steak if the hamburger is in a patty shape. We generate the new plan by performing the constant-for-constant substitutions steak23→hamburger26, onion24→onion27, and oil27→oil25 in steak-and-onions1. Executing the new plan involves executing the events of the transformed episode at the prescribed times, corresponding to this "recipe":

> Slice an onion while heating the broiler for one minute. Put burger under broiler for 4 minutes. When burger has been in for 3 minutes, put frying pan on high heat, adding 1 T oil. Add onions to pan, and flip burger under broiler. Stir onions occasionally for 3 minutes, then remove from pan and remove burger from broiler. Serve together.

We project the facts that the hamburger and onions will be cooked, the onion will be in ring shapes, and the cooking will be a success.

Plan recognition

Part of case-based planning is the assimilation of experience into a usable memory structure. In the steak-cooking case, the relationships between the actions and goals are not explicit; ascribing actions to goals (making the relationships explicit) is the function of plan recognition— given a set of actions, determine the plan (and goals) that they serve. In COOKIE, this is done to a very limited degree, as this conflicts with the philosophy of reasoning from complete episodes. We allow the recognition of low-level action aggregates corresponding to subplans that express action groupings common to a variety of plans. These are expressed as subplan-schemas (corresponding to MOPs) which allow us to relate actions and goals within an episode. A subplan schema consists of goals, steps, and type and temporal constraints on the steps. We have, for example, a subplan schema for frying things; this schema has the goal of cooking some food item and these steps: preparing the food, heating a pan, oiling the pan, adding the food to the pan, interacting with the food in the pan, and removing the food from the pan. When COOKIE processes the steak-and-onions episode, the actions having to do with the onion and pan are used to recognize an instance of a general frying by mapping the actions in the episode to corresponding steps in the schema (and the "cooked onion" goal in the episode to the corresponding schema goal). Once we have recognized the frying subplan here, we allow these parts to be used as a separate episode (for subsequent reasoning) with the added assumption that using this episode is based on the assumption that the onion cooking does not interact with the rest of the cooking (which is recognized as a broiling). Since schemas are only used for recognition (not generation), we can afford to be fairly non-restrictive with the steps—describing them in a general way and allowing most to be optional. This recognition allows the frying and broiling to be used either together or separately; furthermore it gives us an instance of two subplans that can be coordinated in a single meal by one cook.

Explanation

Explanation in COOKIE is closely related to projection and generation, as explanations are only generated when an expected state fails to occur—a generated plan fails or a projection proves false

on execution. The generated explanations can come from two sources. One is the failure of any transformation or separability assumptions made. Suppose we generate a plan for cooking onions based on the onion-frying subplan of steak-and-onions. We explicitly know that this plan is based on underlying assumptions of ingredient fungibility and subplan independence, so if the plan fails, we can assign ingredient difference and subplan dependence as candidate explanations. The other source of explanations is the modification of explanations from other cases (as in SWALE [Kass and Leake, 1988]), which can be used to choose among or make more specific the candidate explanations. The explanation mechanism (*i.e.*, attributing unexpected behavior to causes) in COOKIE is quite simple, but it allows explanation in the absence of a detailed causal theory. Although these explanations do not individually provide the predictive power of a causal explanation, combining evidence from multiple cases could lead to an effective set of predictive features without the need of deep understanding.

Failure recovery

Repair in COOKIE is case-based—execution-time repair is based on remembering and adapting repairs to similar problems problems in previous cases. Such repairs are necessary in planning unless we have either complete prescience or the luxury of being able to "undo" back to a choice point when something goes wrong. In COOKIE's domain, we have neither, and often are faced with problems. The steak-and-onions case had no repairs, but we can use two other cases to illustrate how repair works. The first case is fried-burger-and-onions; it is similar to the previous steak-and-onions, but both the burger and onions are fried in the same pan, the onions being added when the burger is flipped. Due to high fat content, the pan is not oiled for the burger, and the fat released in cooking the meat is enough to keep the onions from sticking. The other cooking is one of curried cauliflower, prepared by stir-frying the cauliflower in 1 T oil, then adding the curry spices for the last couple of minutes of cooking. When it was prepared, the addition of the spices soaked up all of the oil in the pan, causing everything to stick. This was repaired by adding 1 T oil and mixing.

Suppose we have frying onions as a goal; furthermore, we generate this recipe from the onion-frying subplan in fried-burger-and-onions— as in the other burger-and-onions, we separate this meal into a frying of a hamburger and a frying of onions (based on a non-interfering subgoals) with a number of shared steps that could be separated. The generated recipe is to slice onions, heat pan, add onions, stir, remove. When executed, however, the stir fails as the onions stick to the pan. We use the cauliflower case to repair the plan by adding oil, and note two possible explanations for the failure; some subgoal interaction in the original cooking (which is true here), and variation among onions. It doesn't matter that we cannot tell which is which; whenever we fry onions in the absence of meat, we will be prepared to add oil.

4 The role of execution in a CB planner

The purpose of planning is to produce instructions to be executed by some actor. The instructions are a sequence of primitive actions, including information about timing and how progress should be monitored. The information used in this process by a case-based planner is experiential; we get plans for new situations based on what we did in other situations. The subsequent execution of that plan adds an experience to our knowledge base, as well as providing information for evaluating and improving our ability to plan. In COOKIE, the plans given to the execution monitor are episodic representations of "doing it again"; the execution monitor controls execution, monitors expectations, and provides feedback useful in subsequent planning. In this context, plan generation and execution can be considered as mappings: from an episode to expected behavior for generation,

and from expected behavior to a real episode for execution.

From episode to expected behavior

Plans are generated in a direct case-based planner by adapting episodes— transformations are performed on episodes until they match the goals and constraints specified. Basic assumptions in case-based reasoning are that performing the same actions on the same objects would produce identical results, and there exist transformations that are equivalence preserving in terms of those results—if we perform such transformations on these actions, objects, and results, then perform the actions on the objects, we get these results. The adapted episodes can be seen as "expected episodes", and describe the expected behavior of reproducing the described episode.

From expected behavior to episode

Knowing the expected behavior simplifies execution monitoring to a great degree. The actions and conditions in the expected behavior become observed actions and conditions in the episode when execution is done, as long as all expectations are met. If expectations are not met, the anomalies become part of the new episode, and the planner may be invoked to repair the plan. The role of the execution monitor, then, is to cause actions to be initiated at the expected times (if possible), monitor all of the conditions given in the episode, and signal expectation failures back to the planner for possible repair. Expectation failures are labeled as such in the new episode, which will also include any response to that failure.

Execution in COOKIE

Execution monitoring in COOKIE is provided by DEFARGE[1], a dedicated execution monitor that provides the mechanisms for executing plans, adds episodes to COOKIE's knowledge base, interacts with the real-time repair capabilities of COOKIE, and provides feedback to be used in further planning. DEFARGE provides the mapping from expected to real episodes; it causes plans to be executed by side-effect. Its input is the episodic representation of the plan—an adaptation of a real episode. It converts the propositions in the episode into a set of external actions to be directed; starting events, terminating events, and testing conditions, which are assigned expected times based on the times in the episode. In execution, DEFARGE gives instructions and receives information from the external cook. As events terminate and condition tests are reported, the appropriate propositions (with the actual time information) are added to the episode description. If a test results in an anomaly (any expectation failure), the information is forwarded to the planner which can prescribe repairs to be executed and associated with the anomaly.

Since DEFARGE can extract the appropriate tests and expectation from what it is given by the generator, the amount of reasoning done by the monitor is limited. This is by design—the role of the monitor is to extract the necessary information from the plan, interact with the user and planner, do the necessary accounting (keeping track of time, propagating constraints), and convert the execution trace to a usable episode. The relative simplicity is enabled by the detailed information in the episodic representation of the generated plan.

[1] Named for the notorious Mme. Defarge, who monitored quite a few executions [Dickens, 1859].

5 Conclusions

We do not claim that direct use is the only way to use experiential knowledge; it is, however, one that can be used with minimal domain knowledge and computational overhead. It provides a computational framework for reasoning in domains where we don't have detailed, well structured information and causal models. There is strong evidence that human memory is at least partially organized around episodes, and that experience can be used in the future in ways unimagined at the time.

We have only discussed one role for abstraction—in plan recognition we use subplan schemas to ascribe actions to particular goals and to separate out parts of episodes for later use. Most CBR systems, by contrast, have abstraction hierarchies as a central feature of the work. In HYPO, for example, legal cases are viewed at a number of different levels of abstraction [Ashley, 1988]. In CHEF (as in most CBR systems), allowable transformations are implicit in the abstraction hierarchy for objects (allowing transformation between objects that share an ancestor in the isa hierarchy). We do not dispute the usefulness of such abstractions; we simply claim that they are inadequate. Unless we have precomputed all useful abstractions, direct case manipulation is still necessary. Similarly, unless we can somehow encode transformations as a hierarchy that reflects the context-sensitivity of their applicability, we will need to deal with explicit transformation sets.

References

[Ashley, 1988] Ashley, Kevin D. (1988), Modeling legal argument: reasoning with cases and hypotheticals. Tech. Rept. 88-01, University of Massachusetts Department of Computer Science.

[Dickens, 1859] Dickens, Charles (1859). *A Tale of Two Cities.* Oxford University Press.

[Kolodner, 1987] Kolodner, Janet L. (1987). Capitalizing on failure through case-based inference. In *Proc. of the 9th Annual Conference of the Cognitive Science Society,* pp 691–696.

[McCartney and Sanders, 1990] McCartney, Robert, and Kathryn E. Sanders (1990). The case for cases: a call for purity in case-based reasoning. In *Proc. of the AAAI Spring Symposium on Case-based Reasoning.*

[Hammond, 1988] Hammond, Kristian J. (1988), Case-based planning. In *Proc. of a Workshop on Case-based Reasoning,* pp. 17–20.

[Hammond, 1986] Hammond, Kristian J. (1986), Case-based planning: and integrated theory of planning, learning, and memory. Tech. Rept. YALEU/CSD/RR 488, Yale University Department of Computer Science.

[Kass and Leake, 1988] Kass, Alex M. and David B. Leake (1988). Case-based reasoning applied to constructing explanations. In *Proc. of a Workshop on Case-based Reasoning,* pp. 190–208.

[Riesbeck and Schank, 1989] Riesbeck, Christopher K., and Roger C. Schank (1989). *Inside Case-based Reasoning.* Lawrence Erlbaum Associates, Hillsdale, NJ.

[Schank, 1982] Schank, Roger C. (1982). *Dynamic Memory: a Theory of Learning in Computers and People.* Cambridge University Press.

An Internal Contradiction of Case-Based Reasoning[1]

David B. Skalak[2]
Department of Computer and Information Science
University of Massachusetts
Amherst, MA 01003
SKALAK@cs.umass.edu

Abstract

In a case-based reasoning system, one simple approach to assessment of similarity of cases to a given problem situation is to create a linear ordering of the cases by similarity according to each relevant domain factor. Using Arrow's Impossibility Theorem, a result from social welfare economics, a paradox is uncovered in the attempt to find a consistent overall ordering of cases by similarity that satisfactorily reflects these individual rankings. The implications of the paradox for case-based reasoning are considered.

1. Introduction

1.1 Framework for this Paper

This paper attempts to demonstrate that one underlying model of case-based reasoning ("CBR") is internally contradictory. This model, called the "Simple Model" here, assumes that cases can be ranked in order of their similarity to a current problem situation. The conclusion to be drawn from this work is that the model of CBR used by a system must take care to avoid the assumptions that give rise to the contradiction, or must use an entirely different approach, such as one based on numeric weighting of cases, which has its own well-trodden pitfalls.

The path to this result is somewhat circuitous, for it involves translating a result, Arrow's Impossibility Theorem[3], from social welfare economics to CBR. But first as background, case-based reasoning is briefly reviewed, and the problematic model is then presented. An example of a possible CBR contradiction is given.

The first leg of this circuitous path is the presentation of Arrow's Theorem in its original form. In broad scope, the Theorem yields a contradiction that arises in the following circumstances. A group of people rank their preferences for some set of candidates or goods or some other resource. The goal is to derive an overall ranking for the group of individuals as a whole that consistently reflects the individual rankings. Certain apparently reasonable conditions are placed on the compromise ordering and how it should reflect the individual rankings. The Theorem's conclusion is that it is impossible within certain limitations to create a preference procedure to derive the "social" ordering from the individual rankings.

Arrow's original, surprising result is phrased in terms of an "individual" "preferring" one "social state'' to another. To establish the contradiction for CBR, it

---

[1]This work was supported in part by the National Science Foundation, contract IRI-8908481, the Office of Naval Research under a University Research Initiative Grant, contract N00014-87-K-0238, and a grant from GTE Laboratories, Inc., Waltham, Mass.

[2]Copyright © 1990. David B. Skalak. All rights reserved.

[3]Arrow originally referred to the theorem as the General Possibility Theorem (Arrow 1963). Consistent with common usage, I have called it the "Impossibility" Theorem.

will remain to show that Arrow's Theorem can be duly translated to the realm of CBR and the Simple Model. The term "individual" in the Theorem is translated into "domain factor", "preference" into "ranks as more similar to the current problem case", and "social state" into "case". Translated in this way, the conditions of Arrow's Theorem are shown to apply. Thus the conclusion of the Theorem is yielded: that there exists no function to achieve a consistent overall ranking of cases by each individual factor that appropriately reflects the individual rankings. After this result is established, its implications for CBR are discussed.

1.2 Case-Based Reasoning

Case-Based Reasoning is using past problem-solving episodes to analyze or solve a new problem. If necessary, a previous analysis or solution will be adapted to reflect the differences between the new and old cases.

CBR involves several steps, including: (1) accept a new experience and analyze it in order to retrieve relevant cases from case memory; (2) order the retrieved cases to select a set of "best" cases from which to craft a solution or analysis of the problem case; (3) derive the solution by modifying that of the most similar case or formulate an analysis of the problem case in view of the most "on-point" cases; and (4) optionally, store the newly solved or interpreted case in case memory, adjusting indices into memory as appropriate.

1.3 The Relevance of this Result to CBR

When confronted with a situation where a variety of conflicting factors are at work, one may appeal to CBR. In this circumstance, it is difficult to assign blame or credit for the results of a case to a single factor or even to cluster of them. Cases may be viewed from conflicting vantage points that can point to different results (Ashley 1989b). When viewed from the perspective of one factor, one case may seem to be the most similar; another factor may point to another case as being more salient. Or, in a mixed paradigm system or blackboard architecture that uses a variety of knowledge sources to suggest lists of similar cases, those knowledge sources will probably yield different rankings of cases by similarity. *Cf.* (Rissland and Skalak 1989). According to Arrow's Theorem, duly translated, no collective ranking process that meets its preconditions can take the linear orderings of cases by the individual knowledge sources and combine them into a consistent collective similarity ordering.

In the variety of CBR sometimes called "problem-solving" CBR, one case is usually selected as the most similar, and its solution (plan, diagnosis, *etc.*) is modified to solve the original problem posed. If modifying that case turns into a dead end, the next most similar case may be processed and its solution modified. In the variety of CBR referred to as "interpretive" or "precedent-based" CBR, cases are usually presented in an ordering according to their similarity to the current problem case. Both varieties of CBR require that cases be ranked by similarity in some fashion. That different rankings are available to the system, either by virtue of distinct knowledge sources or contrasting factor vantage points, necessitates collecting these rankings into a single ranking. Applying Arrow's Theorem to a simple CBR model suggests that this attempt to construct an overall ordering may involve internal contradictions. Either no procedure will be able to create such a ranking, or we shall have to give up some of the requirements normally placed on the individual rankings and how the collective ranking is to reflect them.

2. The Simple CBR Model

2.1 Description of the Model

Consider the following simplified model of CBR, here called the "Simple Model" . We are given a set of cases, a set of relevant factors, and a problem case called the

"current fact situation". The relevant factors have been pre-designated to determine case similarity and to serve as indices for retrieval of similar cases from the case knowledge base. In this model, the goal of CBR is to rank the cases in a linear ordering in view of their "overall similarity" to the current fact situation.

The Simple Model's apparent solution is to begin by linearly ordering the cases according to similarity with respect to each individual factor. "Ties" are permitted: two or more cases may be equally similar according to a factor. Once these factor-by-factor rankings are assembled, one could try to combine each factor's ranking of cases into a single ordering by overall similarity. I attempt to demonstrate that it is not possible to produce this overall ranking within certain apparently reasonable constraints.

The objection may be made that this Simple Model is unrealistic --- no current CBR system in practice employs it. Several responses to this objection may be made. (1) Elements of the Simple Model are present in some systems, as discussed in Section 5.2. (2) If the Simple Model is not used in its pure form, it may be useful to say why this obvious approach has been avoided. (3) The relevance of the Simple Model may be seen more realistically in the context of a blackboard implementation of CBR, where a number of knowledge sources are available to rank cases. It is likely that individual knowledge sources would suggest contrasting similarity rankings of cases. One would think therefore not in terms of factors ranking cases, but in terms of knowledge sources (that may themselves consider a variety of factors) ranking cases. (4) The factor-by-factor approach is found elsewhere in artificial intelligence. For a recent example see the analysis of the Independent Credit Assignment assumption by Subramanian and Feigenbaum (1986).

2.2 An Example of a Similarity Contradiction

Suppose we have the following three cases and a current fact situation ("CFS") dealing with whether the law permits a driver to overtake an automobile traveling in the same direction. Let us suppose also we have identified three factors as relevant:

1. type of pavement center line (with values none, dotted, single, and double),
2. visibility (poor, fair, good), and
3. distance from position of car to oncoming car in the opposite lane (in feet).

Case	Line	Visibility	Distance	Result
CFS	single	good	800	?
A	single	poor	900	No
B	dotted	good	1000	Yes
C	none	fair	850	Yes

Then, from the point of view of the --

line factor, **case A** is more similar to the **CFS** than **case B**, and **case B** is more similar to the **CFS** than **case C** (write: ApB & BpC, where the relation p may be thought of as "is preferred to");

visibility factor, **case B** is more similar than **case C**, and **case C** is more similar than **case A** (BpC & CpA);

distance factor, **case C** is more similar than **case A**, and **case A** is more similar than **case B** (CpA & ApB).

A majority of the factors --- two out of three in each instance --- indicate that **case A** is more similar than **case B** and that **case B** is more similar than **case C** (ApB & BpC). By transitivity of the preference relation, a majority of the factors

therefore would say that **case A** is more similar than **case C** (ApC). However, a majority of the factors (visibility and distance) actually rank **case C** as more similar than **case A** (CpA). This is a contradiction. This theoretical contradiction presents a practical quandary for a driver's selection of the most similar case in view of the opposite results of **case A** and **case C.** In **case A** passing is not permitted; in **case C**, it is.

3. Arrow's Impossibility Theorem

The above case-based example is actually a reflection of a voting paradox known for over a century (Nanson 1882). Paradoxes like this one have given rise to research by Kenneth Arrow to investigate the conditions under which the ranking by individuals of preferences for "social choices" (for example, for political candidates) may fail to yield a satisfactory "social" (collective) ranking that reflects the individual preference orderings.

Two conditions are placed on the notion of a preference to ensure its accordance with its common meaning. A *preference* (a) must be transitive, and (b) must permit the comparison of any two choices; that is, for any two choices x and y, (x is preferred to y) or (y is preferred to x), or we are indifferent as between x and y (Arrow 1963).

Arrow's Impossibility Theorem on the collective ranking of social choices by individuals is stated in terms of five conditions that might reasonably be placed on a social preference ordering. These conditions are quoted below directly from (Mueller 1979) citing (Vickrey 1960) and are more transparent than Arrow's original renderings. The exception may be Condition IV, which is stated in a more plausible way by Arrow's original work and I have used that formulation[4] (Arrow 1963). The Theorem denies the existence of a *social welfare function*, defined as "a process or rule which, for each set of individual orderings $R_1,\ldots,R_n$ for alternative social states (one ordering for each individual), states a corresponding social ordering of alternative social states, R". (Arrow 1963, p.23). A *social state* is a catch-all term that would include such diverse persons and objects as political candidates, governmental forms, welfare entitlement schemes and access to certain goods.

The five Preconditions to Arrow's Impossibility Theorem:

I. Unanimity (The Pareto Postulate). If an individual preference is unopposed by any other contrary preference of any other individual, this preference is preserved in the social ordering.

II. Nondictatorship. No individual enjoys a position such that whenever he expresses a preference between any two alternatives and all other individuals express the opposite preference, his preference is always preserved in the social ordering.

III. Transitivity. The social welfare function gives a consistent ordering of all feasible alternatives. [That is, the social ranking must be transitive.]

IV. Range. (Unrestricted domain). Among all the alternatives there is a set S of three alternatives such that, for any set of individual orderings $T_1,\ldots,T_n$ of the alternatives in S, there is an admissible set of individual orderings $R_1,\ldots,R_n$ of all the alternatives such that, for each individual i, $x\ R_i\ y$ if and only if $x\ T_i\ y$ for x and y in S.

---

[4]Mueller's statement of Condition IV posits the existence of a "universal" alternative, which assumption he admits is not crucial (Mueller 1979, p.186).

V. Independence of Irrelevant Alternatives. The social choice between any two alternatives must depend on the orderings of the individuals over only these two alternatives, and not on their orderings over any other alternatives.

Arrow's Impossibility Theorem:
No social welfare function satisfies conditions I through V.

The reader is referred to (Arrow 1963) and (Mueller 1979) for proofs of the Theorem.

Arrow's original rendering of the Range Condition may benefit from a little explication. It requires that "every logically possible set of individual orderings of a certain set S of 3 alternatives can be obtained from some admissible[5] set of individual orderings of all alternatives." (Arrow 1963, p.24). The assumption assures that the collective, overall ranking process is not performed in a way that necessarily rules out some possible collective orderings (Mueller 1979). The purpose of the condition is to ensure that the social welfare function is not biased against certain rankings.

4. The Application of Arrow's Theorem to the Simple CBR Model

To establish the result of this paper, it remains to translate Arrow's Theorem from the realm of individual preferences for social states and social welfare functions into the language of CBR: the language of domain factors, similarity, and case ranking procedures.

Let CFS be a given problem case. Instead of saying that an individual prefers one social state to another, say that a *factor prefers* Case 1 to Case 2 if considering that factor alone would lead one to say that Case 1 is "more similar" than Case 2 to the problem CFS. A factor may rank two (or more) cases as equally similar -- in the language of public choice, the factor is *indifferent to* them. This definition of preference may be seen to satisfy the above two criteria placed on a preference. The translation of "social welfare function" is to the procedure that ranks cases by overall similarity from the factors' similarity orderings.

It can be seen that each of the five conditions is reasonably imposed in the translated setting provided by the Simple CBR Model.

For example, the translated Unanimity Condition would state: if a factor's preference between two cases (*i.e.*, that factor's favoring one case as being more similar than another case to a problem situation) is unopposed by any other contrary preference of any other factor, this preference is preserved in the overall ranking of cases. If each factor says that one case is more similar than another to a problem situation, then the collective ordering must preserve this similarity rating.

The Nondictatorship Condition ensures that no factor has so much importance attached to it that it alone can veto the similarity assessment of the remaining factors.

The Transitivity Condition requires that the collective ranking of cases by similarity be transitive. (This condition appears reasonable at first glance, but may be suspect in some applications.)

The Range Condition is perhaps the most opaque, but seems to provide the "twist" that makes the proofs of the Theorem work. In the CBR realm, it requires that there are three cases (A, B and C) that have the property that in the overall ranking any of the relative similarity rankings of those cases is possible for some conceivable

---

[5] An *admissible* set of orderings is a set that is in the domain of the social welfare function, the process that performs the collective ranking.

individual rankings of the cases. This condition requires that these three cases may be ordered in any way, as long as the individuals rankings require it. The upshot of the Range Condition is that the rankings are unbiased to the extent that they do not rule out the possibility of any relative ordering of the three cases. Any of case A, B or C may be the most on point; any may be the least on point. Their relative similarity depends solely on what the individual factors require.

The last condition, the Independence of Irrelevant Alternatives, can also be seen to reasonably apply to the CBR realm. The overall relative similarity of two cases depends only on each factor's ranking of those two cases and not the factor rankings of other cases. If case A is more similar overall than case B, that result does not depend on the individual factor rankings of any other case.

The power of the Arrow's result for public choice is partly due to the self-evident nature of its preconditions. The conditions retain their manifest attraction when translated to the CBR realm. Once one is assured that the five conditions are met, the Theorem's original proofs apply (duly translated), and we have the

Impossibility Result for Case-Based Reasoning: Under these assumptions, there is no process that performs the collective ranking of cases by similarity that satisfies Conditions I to V.

5. Limitations and Implications of the Result

5.1 Arrow's Assumption of No Cardinal Utilities

There is a somewhat obscured assumption to Arrow's results that is not contained in his statement of the Theorem, but is referred to elsewhere in (Arrow 1963). In his own work on social welfare functions, Arrow has assumed that one does not rely on "cardinal, interpersonal comparisons of utility" to obtain the social ranking (Arrow 1963, p.9). In other words, Arrow posits that one may not assign numerical values to gauge the strength of the individual preferences, and use a numerical social welfare function to yield a collective preference ranking for all the individuals. This assumption is rooted in his expressed desire to avoid the difficulties created by assigning numerical values to personal preferences and the conceptual ambiguity of comparing degrees of preference across individuals.

The upshot of Arrow's assumption for the Simple CBR Model is that the Theorem assumes that we are not using cardinal assessments of case similarity for each factor, and then applying an evaluation function to combine these numerical assessments of similarity into an overall ranking. One such constraint is a proscription upon using cardinal numbers to gauge the degree of each factor's preferences. Only ordinal preferences are permitted by the Simple Model. That is, it can be said whether one case is more similar to than another to the current fact situation, but no numerical value can be used to quantify the relative similarities of the cases to the current problem. In particular, this model would not permit using a numerical evaluation function that weights each factor's cardinal preference for a case and sums those weighted preferences into a real number that reflects the collective preference for that case.

Following Arrow's interdiction against cardinal utilities is reasonable in many CBR domains. Numerical weighting schemes for credit assignment may be avoided in view of a litany of conceptual and pragmatic objections against their use. See, *e.g.*, (Ashley 1989a), (Ashley and Rissland 1988), (Kolodner 1988). A number of these objections are collected in (Ashley 1990): (1) lack of cognitive validity for many domains and domain experts; (2) uncertainty and arbitrariness in weight assignment and lack of an authoritative source of numerical weights, with the consequent disagreement among experts as to specific weights; (3) the failure of static weights to

reflect the context-dependent importance of factors; (4) pitfalls of a premature commitment to a weighting scheme; and (5) reduction of information into a single number entails a representation that loses information that might otherwise be used to advantage.

5.2 Three Weighting Models

Ashley (1989a) identifies three models of factor weighting that underlie current CBR systems: numerical, precedent-based, and heuristic. The consequences for each of these models of the Impossibility Result are discussed below.

One way for a CBR system to avoid the paradox of the theorem is to use numerical functions to assess case similarity. See, *e.g.*, (Kibler and Aha 1987). Numerical weights may be appealed to in domains where an analytic or statistically-based model is accessible. See (Ashley 1989a). In domains where numeric weighting of factors is reasonable, the Nondictatorship Condition of the Theorem may be violated, however. A factor with a sufficiently large weight may determine the outcome of the similarity ranking, regardless of the rankings by lesser weighted factors. However, an evaluation function that is used merely for its ordinal, and not its cardinal, properties --- used just to rank cases by similarity in a total order --- may be subject to the strictures of the Theorem nonetheless.

In some areas of application, models of any sort are simply not available, and one must appeal to previous cases to assess the relative importance of factors (Ashley and Rissland 1988). The lack of some domain models is apparently a classical impetus for the use and development of case-based reasoning techniques in the first place. In such applications of CBR, one must rely merely on the patterns of combinations of factors that have appeared in previous cases to perform credit assignment. It is difficult to identify the factor(s) that determine the results in such instances, and one must avoid embracing a ranking scheme that reduces to a voting arrangement. In such precedent-based schemes where no numeric or heuristic weighting is available, special care must be taken to avoid the siren song of the Theorem. The scheme used must violate at least one of the Theorem's preconditions, explicit or implicit, or be potentially subject to the paradox. In the work of Rissland and Ashley, for example, (a) cases are ordered by clusters of factors and not by individual factors, (b) that ordering is partial and not total, and (c) the Range Condition may not be satisfied.

The appeal to an analytic model provides an intermediate point between numeric weights and the assessment of the importance of factors based on precedent. Where a causal or other analytic model is available, it may be used to rank goals and the features that influence their attainment (Koton 1988), (Bareiss 1987), (Kolodner 1985), (Hammond 1987). See generally (Ashley 1989a). Where factors can be ranked in a preference ordering according to their importance, the Theorem's paradox is circumvented through the violation of the Nondictatorship Condition. The similarity ranking may be dominated by the most important feature. For example, Kolodner's PARADYME system avoids the paradox by using an ordered set of preference heuristics to choose the most useful cases from those retrieved from a previous partial matching (Kolodner 1989). In general, systems that use a model to rank goals (and the factors that influence goal attainment) may avoid the potential for internal contradiction by avoiding the Theorem through the violation of the Nondictatorship Condition.

6. Conclusion

CBR is inherently subject to a dilemma. On one hand, it is problematic to assign numerical values and weights to similarity preferences, and thereby hide the resulting "value" judgements in trading off disparate and fundamentally

incommensurate factors. On the other hand, one must avoid embracing a symbolic or implicit weighting scheme that can be construed as a voting procedure of the sort here presented with its attendant paradoxes.

References

Arrow, K. J. (1963). Social Choice and Individual Values. Cowles Foundation for Research in Economics at Yale University. New Haven, Yale University Press.

Ashley, K. D. (1989a). "Assessing Similarities Among Cases." Proceedings of the Second DARPA Case-Based Reasoning Workshop, May 1989, Pensacola Beach, FL.

Ashley, K. D. (1989b). "Toward a Computational Theory of Arguing with Precedents: Accommodating Multiple Interpretations of Cases." In Proceedings of the Second Annual Conference on AI and Law, June 1989, Vancouver, B.C.

Ashley, K. D. (1990). Modelling Legal Argument: Reasoning with Cases and Hypotheticals. Cambridge, MA, MIT Press (in press).

Ashley, K. D. and Rissland, E.L. (1988). "Waiting on Weighting: A Symbolic Least Commitment Approach." Proceedings AAAI-88, American Association for Artificial Intelligence, August 1988, Minneapolis.

Bareiss, E. R., Porter, B.W., and Wier, C. C. (1987). "Protos: An Exemplar-Based Learning Apprentice." In Proceedings of the Fourth International Workshop on Machine Learning, pages 12--23, June 1987, University of California at Irvine.

Hammond, K. J. (1987). "Explaining and Repairing Plans that Fail". In Proceedings IJCAI-87, International Joint Conferences on Artificial Intelligence, Inc., August 1987, Milan.

Kibler, D. and Aha, D.W. (1987). "Learning Representative Exemplars of Concepts: An Initial Case Study." Proceedings of the Fourth International Workshop on Machine Learning, pages 24--30, June 1987, University of California at Irvine.

Kolodner, J. L. and Simpson, R. L., and Sycara-Cyranski, K. (1985). "A Process Model of Case-Based Reasoning in Problem Solving." In Proceedings IJCAI-85, International Joint Conferences on Artificial Intelligence, Inc., August 1985, Los Angeles.

Kolodner, J. L. (1988). "Retrieving Events from a Case Memory: A Parallel Implementation." Proceedings of the DARPA Case-Based Reasoning Workshop, May 1988, Clearwater Beach, FL.

Kolodner, J. L. (1989). "Judging Which is the `Best' Case for a Case-Based Reasoner." Proceedings of the DARPA Case-Based Reasoning Workshop, May 1989, Pensacola Beach, FL.

Koton, P. (1988). "Reasoning about Evidence in Causal Explanations." In Proceedings of the First DARPA Case-Based Reasoning Workshop, May 1988, Clearwater Beach, FL.

Mueller, D. C. (1979). Public Choice. Cambridge Surveys of Economic Literature. Cambridge, England, Cambridge University Press.

Nanson, E. J. (1882). Transaction and Proceedings of the Royal Society of Victoria. **19**: 197-240.

Rissland, E. L. and Skalak, D. B. (1989). "Combining Case-Based and Rule-Based Reasoning: A Heuristic Approach." Proceedings IJCAI-89, International Joint Conferences on Artificial Intelligence, Inc., August 1989, Detroit.

Subramanian, D. and Feigenbaum, J. (1986). "Factorization in Experiment Generation." In Proceedings AAAI-86, American Association for Artificial Intelligence, August 1986, Philadelphia.

Vickrey, W. (1960). "Utility, Strategy, and Social Decision Rules." Quart. J. Econ. **74**(Nov.): 507-35.

Qualitative Reasoning about the Geometry of Fluid Flow

Hyun-Kyung Kim

Qualitative Reasoning Group
Beckman Institute, University of Illinois

Abstract: Understanding the interaction between dynamics and geometry is crucial to capturing commonsense physics. This paper presents a qualitative analysis of the direction of fluid flow. This analysis is dependent on qualitative descriptions of the surface geometry of rigid bodies in contact with the fluid and a pressure change in fluid. The key problem in designing an intelligent system to reason about fluid motion is how to partition the fluid at an appropriate level of representation. The basic idea of our approach is to incrementally generate the qualitatively different parts of fluid. We do this by dynamically analyzing the interaction of geometry and pressure disturbance. Using this technique, we can derive all possible fluid flows.

1 Introduction

Understanding the interaction between dynamics and geometry is crucial to capturing commonsense physics. Without spatial reasoning, dynamics cannot fully explain the physical world. For example, applying the same force to different points on an object can cause dramatically different behaviors. Without geometric information, these behaviors would be difficult to predict.

Unfortunately, the general spatial reasoning problem is intractable. Thus, recent research has focused on more constrained problems such as motion in limited domains [2,3], mechanical mechanisms [7,8] and fluid ontologies [1,6]. The studies dealing with mechanical mechanisms and motion focus only on rigid objects, ignoring the motion of fluid. In addition, the fluid ontology research is insufficient to fully explain fluid behavior. Two basic approaches to fluid ontology are *contained-stuff* ontology and *piece-of-stuff* ontology [1,6]. Neither of these approaches suffices to explain the geometry of fluid motion. Suppose we want to explain the motion of the gas in a piston when the valve in the middle of the right side is open and the pressure inside the piston is greater than the pressure outside. Since *contained-stuff* ontology treats the gas in the piston as one object, it is impossible to reason about different flow directions in the various parts of the piston. Similarly, it appears to be impossible to consider the motion of every piece of the gas. People do not seem to use either ontology to explain this problem. However, people can reason about the gas near the top surface moving downward to the right and the gas near the bottom surface moving upward to the right, etc.

This paper presents a technique for reasoning about the direction of fluid flow in two-dimensions using incremental generation of the *places* in space. We extend the work of [5,8,3] on places—from the rigid body domain onto the fluid domain. Since fluid motion is determined by the pressure difference and the geometry of surface contact with the fluid, we assume qualitative descriptions of these two terms as input. The theory predicts an equivalence class of places that are created based on the qualitative behavior of the fluid. In addition, it describes the flow in each of these places.

Section 2 presents the theory for reasoning about flow direction in qualitative and geometric terms, given a pressure change and surface geometry. The fluid are partitionied so each part has the same qualitative fluid motion. Section 3 explains how envisionments qualitatively simulate the behaviors of fluid in each part. In section 4 we summarize our results and discuss possible extensions to our theory.

2 A Qualitative Theory of Fluid Motion

The key problem in commonsense reasoning about fluid motion is how to partition the fluid at an appropriate level of representation. Our approach partitions the fluid into a set of *places* by reasoning about pressure wave propagation and geometry. These two factors determine the flow direction. We show how to decompose space incrementally into places.

2.1 Qualitative Direction

In spatial reasoning, the concept of direction is essential in describing the position, force, and motion [8]. We assume a single global reference frame. This reference frame can be translated but not rotated. In our theory, direction is represented by a qualitative vector [8] and qualitative vector arithmetic is used to compute the directions. In our 2D space representation, the first and second components of a qualitative vector represent the qualitative direction along the x-axis and y-axis, respectively. To represent the x-axis direction, we use "+" for "right" and "−" for "left" and "0" for center. For the y-axis, "+" is used for "up" and "−" for "down" and "0" for center. For example, (−−) indicates the vector lies to the lower left of some reference frame.

Definition 1 (Inversion) *Inversion*(v) is the qualitative vector v rotated by 180 degrees.

Definition 2 (Open-Half-Plane) *Open-Half-Plane*(v) are the vectors whose vector dot product with v is "+".

2.2 Rigid Object Representation

Rigid objects are represented by their surfaces in contact with the fluid. In our 2D space, a surface is represented as a line segment. For each surface, we represent the direction of the line segment as the position of one end-point relative to the other. (We define end-points as the points where the line segment meets the neighbors.) In general, the relative position can be defined for any two points:

Definition 3 (Relative-Position) *Relative-Position*(p1,p2) is the qualitative vector which represents the direction from point p2 to point p1.

Consider a surface with end-points p1 and p2, which are connected to other adjacent surfaces. The direction of the surface is represented by Relative-Position(p2,p1).

Information about the relative position can be propagated using transitivity:

Law 1 (Transitivity of Relative-Position) For any points p1,p2, and p3, Relative-Position (p1,p3) is computed by adding given values Relative-Position(p1,p2) and Relative-Position (p2,p3).

The surface normal represents the direction which points from the surface into fluid.

Definition 4 (Surface Normal) *Surface-Normal*(s) is the qualitative vector which represents the surface normal of the surface s.

Definition 5 (Surf-Rel-Pos) For any two adjacent surfaces s1 and s2 whose end-points are (p1,p2) and (p2,p3), *Surf-Rel-Pos*(s1,s2) represents Relative-Position(p1,p3). This represents the relative direction of two adjacent surfaces.

2.3 Fluid

Unlike a solid, a fluid moves and deforms continuously as long as a pressure difference exists. Its shape is determined by the container. These properties of fluid make them difficult to individuate in a reasoning system. In general, people do not reason about the individual molecules of fluids, but rather they focus on the collection of molecules within fluids.

Pressure Wave Propagation (PWP): When a pressure disturbance occurs in a compressible fluid, the disturbance travels with a velocity of sound. For example, if we throw a stone into the pond in rest, we can see the circular wave-fronts on the surface are diverging from the source. If the disturbance is due to the lower pressure, then an *expansion* wave is propagated. If it is due to the higher pressure, then a *compression* wave occurs. The pressure wave moves from the source toward the wave-fronts and it is perpendicular to the wave-fronts. As the pressure wave is propagated through a still fluid, the fluid properties (i.e., pressure, temperature, and density and so on) change and it start to move. As a compression wave is propagated, the fluid molecules have a velocity which has the same direction as the wave propagation. On the other hand, when an expansion wave travels, the fluid has a velocity which has the opposite direction of the wave (i.e., toward the

source of the disturbance). The induced velocity of the fluid by wave propagation is much slower than the wave propagation.

Definition 6 (Prop-Constraint) Suppose a surface s is in contact with the fluid. *Prop-Constraint* (*s*,*d*) is *true* when PWP is prevented in direction *d* near s.

Law 2 (Surface-Constraint) Suppose s is in contact with the fluid and its surface normal is *sn*. Then the pressure wave cannot propagate from the surface to the fluid. Thus for every *d* which belongs to Open-Half-Plane(*sn*), Prop-Constraint(*s*,*d*) is *true*.

Continuous Change (CN): We assume the flow is smooth and steady (*laminar*). When flow is not laminar but fluctuating and agitated (*turbulent*), it is impossible to explain the behavior. Even in fluid mechanics, no general analysis of fluid motion in turbulence yet exists and there may never be. People also have difficulty explaining the direction of the turbulent flow. In laminar flow, the changes of properties are continuous. To support laminar flow, we assume the surface is smooth and the changes in surface are not abrupt.

2.4 Place Generation

In FROB [3], given a geometric description of the surface, the places needed to envision the possible motions of a ball are generated by the constraints of geometry of the surface and gravity. Since the gravity constraint is the same everywhere, space can be divided without regard to the neighbors. However, since a direction of PWP can keep changing by the surface geometry of rigid body as it propagates, the place cannot be generated without considering interaction between these two. Even though two given spaces have the same geometry, they can be partitioned in completely different ways with different directions of pressure wave.

In fluid motion, qualitatively different parts have different fluid directions since the pressure wave arrives from different directions. Thus *places* in our reasoning problem should be distinguished by difference of the direction of pressure wave in each part. Continuous interaction of pressure wave and geometry suggests our place generation should be incremental as the pressure wave propagates from the source of disturbance.

Definition 7 (Place) A *Place* is defined by its boundaries (i.e., left, right, up, and down) and the direction of the pressure wave and the direction of the induced velocity. Given a place P, *Pres-Wave*(P) returns the the direction of pressure wave in P. The boundaries represent the adjacent places or surfaces of a given place.

Definition 8 (Place) *Place*(s) maps from a surface s to the *place* in which s is a boundary.

Law 3 (Connectivity of Place) Given two adjacent surfaces s1 and s2, Place(s1) and Place(s2) are also adjacent. Furthermore, since the relative direction between the two surfaces is Surf-Rel-Pos (s1,s2), Place(s1) is oriented in the same direction.

As the law of *Connectivity of Place* shows, to represents the connectivity of places, the numeric information is not used. It is represented by the relative position between the places in qualitative terms. For example, in in Figure 1 Place(s5) is located to the right of and above Place(s3) since Surf-Rel-Pos(s5,s3) is $(-+)$.

In our approach, places are generated incrementally from the initial pressure change to the direction of PWP. For example, in Figure 1, when portal becomes open, the expansion wave is propagated from the outside to s1 and s2 first since they are close to the outside. Then places are generated around these surfaces. After that, places near s3 and s4 are generated as the pressure wave keeps traveling. Figure 1 graphically shows this incremental generation.

Since the surfaces of a rigid body can be the only explicit boundaries of the fluid from the input, our approach first generates the places near the surfaces by propagating the pressure wave across the pairs of adjacent surfaces. The places of the space which are not bounded by the surface are not generated at first since it is impossible to trace every point and give the boundary of the place. But as places around the surfaces are generated, the whole space can be covered by the property of fluid, CC. For example, in Figure 1, when places near s1, s2, s3, s4, s5,and s6 are generated, by

Figure 1: Incremental Place Generation in Piston-Cylinder

An oval represents the a place generated. The locations of ovals represent the connectivity of places.

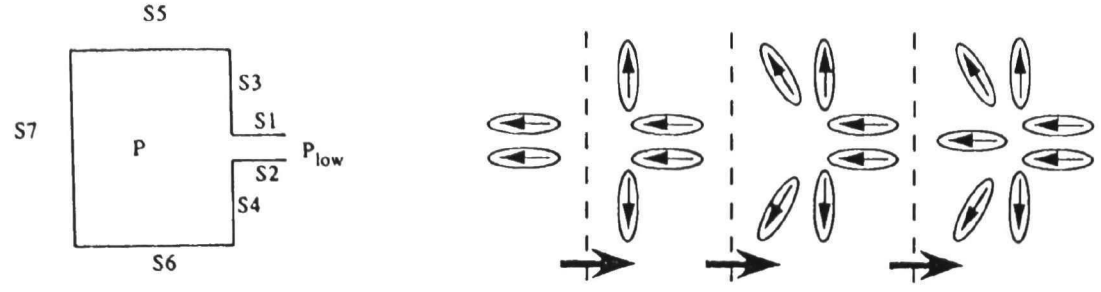
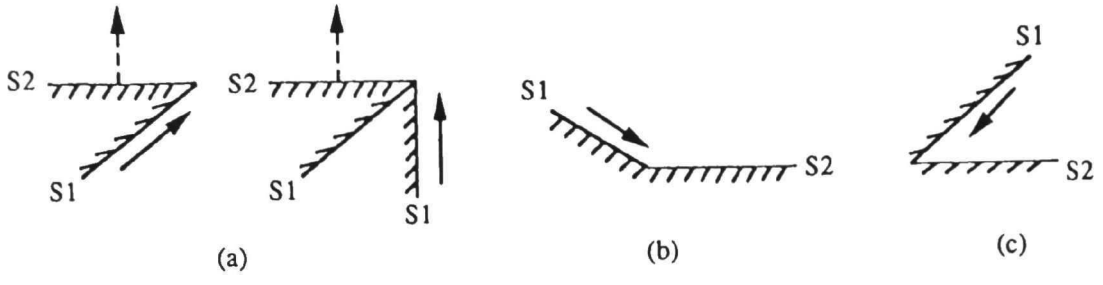


Figure 2: Examples of Forward Propagation when Surface-Normal(s2) = (0+)



CC, a place whose Pres-Wave is (−0) is generated between them .

To propagate a pressure wave across the pairs of adjacent surfaces, the first step is to determine the propagation order between the adjacent surfaces. Given a newly generated place, our system checks how the pressure wave can propagate toward adjacent surface.

Definition 9 (Forward-Propagation?) Suppose s1 and s2 are two adjacent surfaces and their end-points are (p1,p2) and (p2,p3). If the Pres-Wave(Place(s1)) belongs to Open-Half-Plane(Relative-Position(p2,p1)), then *Forward-Propagation?*(s2, s1) is *true*. Otherwise, it is *false*.
As Figure 2 shows, when Forward-Propagation?(s2, s1) is true, we can infer the source of disturbance is not closer to s2. Thus the following law is introduced.

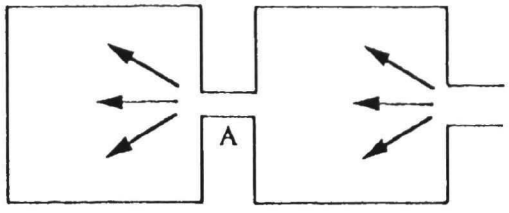
Law 4 (Forward Propagation) Suppose Forward-Propagation?(s2, s1) is *true* and the end-points of s1 and s2 are (p1,p2) and (p2,p3). Then s2 belongs to *next wave front* of s1 if Pres-Wave(Place(s1)) belongs to Open-Half-Plane(Surface-Normal(s2)) (*blocked*) (Figure 2a) or if s2 is not blocked and Pres-Wave(Place(s1)) belongs to Open-Half-Plane(Relative-Position (p3,p2)) (*further*) (Figure 2b). s2 belongs to *same wave front* of s1 if s2 is not blocked and Pres-Wave(Place(s1)) belongs to Open-Half-Plane(Inverse(Relative-Position(p3,p2))) (Figure 2c).

Law 5 (Further Propagation) Suppose Forward-Propagation?(s2, s1) is *true*. If s2 is *further* than s1 from the source, then Pres-Wave(Place(s2)) from the source belongs to Open-Half-Plane (Pres-Wave(Place(s1))).

This law shows if PWP is not blocked, then it smoothly changes the direction across the adjacent surfaces.

A pressure wave travels from the source of disturbance to the all direction unless it is blocked by the surface of the rigid body. Unless the direction of PWP is changed by any surface, then Pres-Wave of any point can be simply inferred as direction from the source to that point. However, as the wave travels, a new source can be generated by geometry of the surfaces. Figure 3 shows how a

Figure 3: New Source



new source is generated when wave arrives at point A from the source.

We identify the cases to generate the new source.

Law 6 (New Source) Suppose Forward-Propagation?(s2, s1) is *true* and the end-points of s1 and s2 are (p1,p2) and (p2,p3). Then there are two cases to generate a new source: (1) *Blocking* -If s2 is *blocked*, then new source is generated around p2. (2) *First-Moving* - If s2 is *further* than s1 from the source, then new source might be generated in Place(s1). Since the fluid in Place(s1) starts to move earlier than that of s2, this may cause a pressure disturbance.

Once the next surface to be propagated is chosen and a new source is identified, Pres-Wave(Place (s2)) is determined by the relationship between s1 and s2, say, whether s2 is in the next wave front or the same wave front and a new source might be generated. In the case of *Blocking*, Relative-Position (p3,p2) becomes Pres-Wave(Place (s2)) since a new source of disturbance is generated near p2. In case of *First-Moving*, PWP by the original and the newly generated source should be considered. If a pressure wave can arrive at s2 without the blocking by surface, the relative direction from the source of disturbance to the surface s2 can be computed by adding that of s1(i.e., Pres-Wave(Place(s1))) and Relative-Position(p3,p2) by the law of *Transitivity of Relative-Position*. Since our approach is based on the qualitative information, every possible inference is made. Thus Relative-Position(p3,p2) and Pres-Wave(Place(s1)) + Relative-Position(p3,p2) are possible directions for Place(s2). In the case of *Same wave front*, at least the parts of s2 which are closer to s1 have the same Pres-Wave as s1. Thus places of s1 and s2 are *merged*. As we mentioned previously, during generation of the places based on this, new places may be generated by the CC.

2.5 Inferring Propagation in Backward Direction

Since our approach propagates the pressure wave across the connected surfaces and divides the space based on the places near the surface, it may not suffice given a more complicated geometry. For example, in Figure 4 when Place(s1) is generated, a new place cannot be generated any more since there is no surface adjacent to s1 in the forward direction of PWP. By the same reason, new place cannot be generated after Place(s4).

However, by the reverse inferencing of the forward PWP, this problem can be solved. As we can expect, this reverse inference may bring out ambiguities. Since our technique does not use any metric information, several possibilities can be introduced. But using some constraints due to the characteristics of fluid, we can eliminate many ambiguities.

Definition 10 (Backward-Propagation?) Suppose s1 and s2 are two adjacent surfaces and their end-points are (p1,p2) and (p2,p3). If the Pres-Wave(Place(s1)) belongs to Open-Half-Plane (Inverse(Relative-Position(p2,p1))), then *Backward-Propagation?*(s2,s1) is *true*.

If Backward-Propagation?(s2,s1) is *true*, then Forward-Propagation?(s1, s2) is *true*. Thus s1 belongs to the next wave front or the same wave front of s2. We identified how to infer whether s2 belongs to previous or same wave front of s1 from the geometric analysis for all possible cases.

Figure 4: Complex Geometry

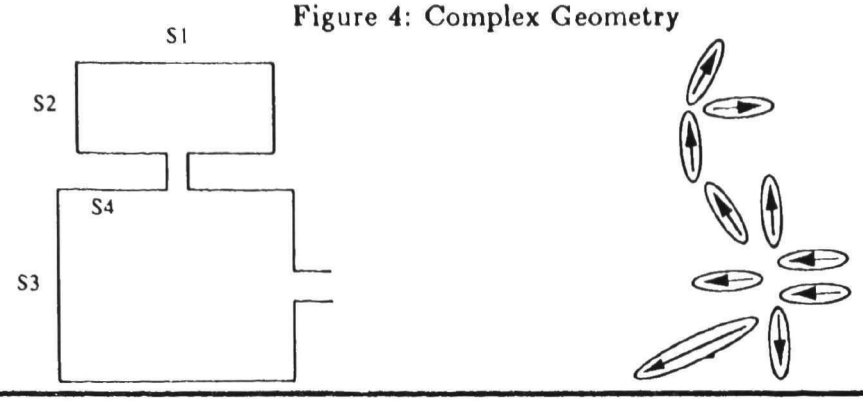
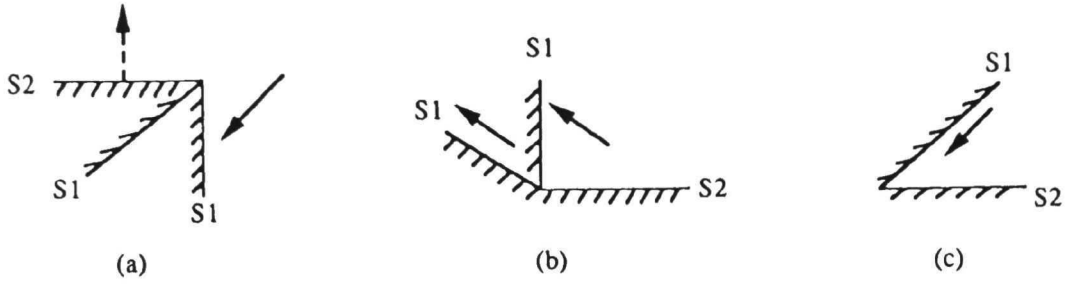


Figure 5: Examples of Backward-Propagation?((0+),s1)



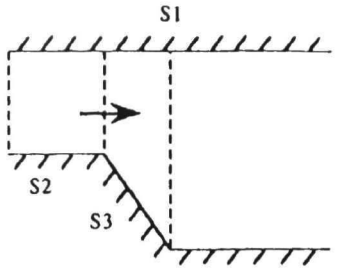
Law 7 (Backward Propagation) If BackwardPropagation?(s2,s1) is *true*, the relationship between s1 and s2 in propagation of a pressure wave is: (1) *same wave front* - Pres-Wave(Place(s1)) belongs to Open-Half-Plane(Inverse(Surface-Normal(s2))); In addition, if Relative-Position (p1,p2) is equal to the Pres-Wave(Place(s1)), then this geometry can be the reverse of the *Blocking* in forwarding propagation (Figure 5a), (2) *previous wave front* - unless s2 is *same wave front* (Figure 5b).

In case of *same wave front*, Pres-Wave(Place(s2)) is computed by adding Pres-Wave(Place(s1)) and Relative-Position(p3,p2). For the reverse of the *Blocking*, Pres-Wave(Place(s2)) can be any d which belongs to Open-Half-Plane(Surface-Normal(s1)) and is not constrained can be Pres-Wave(Place(s2)). In the case of *previous wave front*, we cannot compute the possible directions but can give constraints which filter the illegal ones.

Law 8 (Source-Constraint) Suppose Backward-Propagation?(s2, s1) is *true* and the end-points of s1 and s2 are (p1,p2) and (p2,p3). Then, (1) pressure wave cannot propagate from p2 to s2. Thus for every d which belongs to Open-Half-Plane(Relative-Position(p3,p2)), Prop-Constraint(s2, d) is *true*. (2) pressure wave cannot propagate from s1 to s2. Thus by the law of *Further Propagation*, for every d which belongs to Open-Half-Plane(Inverse(Pres-Wave(Place(s1)))), Prop-Constraint (s2,d) is *true*.

By the inferring in reverse order, it is not easy to determine the direction of a possible pressure wave; there may be several possibilities. When there are several possibilities for one problem, people tend to eliminate inadequate ones by constraints to get the final solutions. By giving the *Surface* and *Source* constraints, we can filter out the illegal ones. For example, in Figure 5b, if we apply these two constraints, the only possible directions for Pres-Wave(Place(s2)) are (-0) and $(-+)$. In case of c, no direction is left after filtering out, which means Pres-Wave(Place(s1)) cannot have the direction of $(-+)$. Like in c, even if the illegal places generated by reverse inference by the lack of information, they can be filtered out later.

Figure 6: Flow and Surface Interaction



3 Envisioning Flow direction

Given an external disturbance of pressure, the space of interest is incrementally divided by connected places. Our system computes every possible combination of the places. Then the fluid in each place starts to move in the direction of a pressure wave if it is a compression wave or in the opposite direction of a pressure if it is an expansion wave. By the connectivity of the places, we can predict the next place where the fluid will go. For example, if the induced velocity of a place is (+−), then the fluid in that place will flow into the places where are located to the right or down from the place.

When the moving flow comes to the place, then the induced force by the interaction of the flow and surface of the rigid body may be applied to the flow. Figure 6 shows an example of this. Arrows represent the direction of the induced velocity. When the fluid in Place(s2) arrives at Place(s3) with the velocity (+0), it keeps going to that direction. However, the area near the surface s3 change and has less molecule of the fluid compared to the other parts of Place(s3).

Compared to the pressure disturbances in previous section, the influence of disturbance of flow and geometry is small and local since as soon as it happens the flow in that place changes the direction by the induced force. Thus its disturbance is diminished. But even if the induced force is applied to the moving flow, the flow does not immediately change the direction to the direction of the applied force since the flow already has the momentum. Since this effect is local to the flow in that place, it generates a *local* place inside the place. Its effect assumes to be limited inside of the place. Even though we can infer this region exists inside of the place, it seems to be impossible to explicitly give their boundary since its effect keeps diminishing and the interaction between the flow close to that region and that disturbance keeps changing.
Since the local place is also generated by pressure change, two kinds are possible:

Definition 11 (N-Local-Place) Suppose P represents Place(s). If the pressure adjacent to s is lower than the other part of inside of P, then *N-Local-Place*(P,s) is generated near s.

Definition 12 (P-Local-Place) Suppose P represents Place(s). If the pressure adjacent to s is higher than the other part of inside of P, then *P-Local-Place*(P,s) is generated near s.
We identified the interaction between flow and geometry as follows:

Law 9 (Pulling) Suppose P represents Place(s) and the flow with the velocity v is entering the place P. If Surface-Normal(s) belongs to Open-Half-Plane(v), then N-Local-Place(P,s) is formed. This N-Local-Place gives the force in direction of Inverse(Surface-Normal(s)) to the flow in P (i.e., it pulls the flow into the surface).

Law 10 (Pushing) Suppose P represents Place(s) and the flow with the velocity v is entering the place P. If Surface-Normal(s) belongs to Open-Half-Plane (Inverse(v)), then P-Local-Place(P,s) is formed. This P-Local-Place gives the force in direction of Surface-Normal(s) to the flow in P (i.e., it pushes the flow into the surface).
Pushing happens since the fluid molecules hits the wall and those collisions increase the pressure near the wall.

Thus fluid direction in each place can be envisioned by starting from its original place and traveling into the adjacent places with possible changing of its direction due to the surface interaction. The flow will be stop if it goes to equilibrium after moving.

4 Discussion

This paper presents a theory of geometry of fluid flow in two-dimensional space. Given qualitative descriptions of geometry and a pressure change in fluid, we can determine the possible directions of fluid motion. The interaction between the surface geometry of rigid body and pressure wave propagation is identified in our theory. This idea is being implemented.

We have only dealt with the velocity change of fluid here. We plan to expand our theory to have a complete theory for reasoning about fluid. Reasoning about the other important properties of fluid, such as pressure, temperature, and density so on is left as future work. What we hope to analyze eventually is a real system, such as internal combustion engine, which should be explained by tightly integrating dynamics and kinematic of rigid bodies and fluid. Our theory for analyzing the directions of fluid flow is one step towards that goal.

5 Acknowledgements

I would like to thank Ken Forbus for his guidance. Thanks to Janice Skorstad, John Collins, and Dennis DeCoste for proofreading and comments. This research was supported by the Office of Naval Research, Contract No. N00014-85-K-0225.

References

[1] Collins, J. "Reasoning About Fluids Via Molecular Collections", Proceedings of AAAI-87, July, 1987

[2] deKleer, J. "Qualitative and Quantitative Knowledge in Classical Mechanics", TR-352,MIT AI Lab, Cambridge,MA., 1975

[3] Forbus, K. "A Study of Qualitative and Geometric Knowledge in Reasoning about Motion", TR-615,MIT AI Lab, Cambridge,MA, 1981

[4] Forbus, K. "Qualitative Process Theory", *Artificial Intelligence*, **24**, 1984

[5] Forbus, K., Faltings, B., and Nielsen, P. "Qualitative Mechanics: A Framework", UIUCDCS-R-87-1352, University of Illinois, 1987

[6] Hayes, P. "Naive Physics 1: Ontology for Liquid" in Hobbs, J. and Moore, B. (Eds.), *Formal Theories of the Commonsense World.* Ablex Publishing Corporation, 1985

[7] Joskowicz, L. "Shape and Function in Mechanical Devices" Proceedings of AAAI-87, July, 1987

[8] Nielsen, P. "A Qualitative Approach to Mechanical Constraint" Proceedings of AAAI-88, August, 1988

[9] White, F. *Fluid Mechanics*, McGraw-Hill Book Co., 1979

Is There a Default Similarity Distance for Categories?

Yaakov Kareev and Judith Avrahami
The Goldie Rotman Center for
Cognitive Science in Education
School of Education
The Hebrew University of Jerusalem

Abstract

How do people decide whether or not an item belongs to a new category, the variability of which they do not know? We postulate that people have a default similarity distance (DSD) which they use when no other information about the variability of a category is available. To test our claim, subjects were asked to tell how they would instruct a being from another world to distinguish members of a category, by showing pictures. The categories were from different levels thus differing in variability. For highly variable categories subjects tended to present multiple positive instances (thus indicating their extraordinary variability), whereas for narrow categories they tended to present negative instances (thus explicitly delimiting them). These results indicated that a norm, relative to which additional information is supplied, lay in between. Indeed, there was a level at which subjects apparently relied on DSD, finding it sufficient to show but a single exemplar of the category. This happened with basic-level categories for 8th graders and adults and with subordinate categories for 2nd graders, thus demonstrating a developmental trend in what is considered a normal standard category.

Dealing with the classification of new items into categories, researchers have focused mainly on what it is that a new item is compared with to decide whether it belongs to a particular category or not. That entity is variously claimed to be a prototype (Rosch, 1973; Rosch & Mervis, 1975), a set of all properties of members of the category (Hayes-Roth & Hayes-Roth, 1977), or a collection of all exemplars of the category hitherto encountered (Medin & Shaffer, 1978). All theories agree, however, that the new item need not be identical to any of the above; it just has to be sufficiently similar or, in Medin and Barsalou's (1987) terms "above a certain threshold" of similarity.

It is clear that there is no single value which defines "sufficient similarity," since lower similarity is allowed between items of a higher level category than between items of a lower level category. Thus, an object has to

be more similar to other bulldogs to be considered a bulldog than to other animals to still be considered an animal. Indeed, Fried and Holyoak (1984) suggest that the representation of a category includes both a mean value of the category and an indication of the density of its exemplars in a feature space. This density can be viewed as the variability or similarity distance allowed and expected between exemplars of the category.

To better understand classification we still need some explanation for situations in which classification is based on a single item or in which categories are formed with no feedback. We suggest that people have a notion of a "proper" distance or "plausible" similarity which serves as the threshold mentioned above. It is a default similarity distance (DSD), used when no information about the variability of the category or about its neighboring categories is available. We expect it to be some middle value, close to the mean similarity distance of categories known to the subject. Whatever the initial similarity distance assumed, it is continuously updated following subsequent encounters with exemplars which are known to belong to the category in question and ones that do not.

To test our hypothesis that people have such a DSD we had our subjects tell how they would teach a creature from another world to identify members of a certain category by showing it pictures. If there is no DSD one would expect subjects to provide for <u>any</u> category not only a representative member of it but also some indication of its variability, or allowed similarity distance. If, on the other hand, subjects take DSD into account and assume others to share it, they would see a lesser need to indicate the variability of categories for which DSD is more appropriate; for those categories, they may consider it sufficient to present but one typical exemplar.

The argument, then, goes as follows: for categories whose similarity distance is close to that denoted by DSD, subjects will be more likely to be

satisfied with a single representative exemplar of the category than for categories where DSD is inadequate. In the latter case subjects will be more likely to provide additional information to indicate the variability of the category.

How can the variability of a category be indicated? To indicate that the variability of a category is greater than the expected value, multiple, various exemplars belonging to that category can be used. When the variability of a category is narrower than the expected value, negative exemplars can be used - items which do not belong to the category in question but would belong if it were a category with the default distance. To insure a wide range of variabilities, we included categories of different levels: basic-level, superordinate, and subordinate.

To find out whether DSD changes with age, we employed subjects of different ages.

Method

Design. The study had a two-way factorial design with the variables of age and level of category.

Subjects. The sample consisted of 122 2nd graders (aged 7(3) to 8(2)) and 187 8th graders (aged 13(3) to 14(2)) from middle-class neighborhoods and 110 undergraduate students attending an introductory course in statistics at the Hebrew University.

Materials. The following items were used in the study: a) Superordinate categories: animal, plant, means of transportation; b) Basic-level categories: dog, bird, tree, mushroom, car, boat; c) Subordinate categories: bulldog, dachshund, wagtail, oal, cypress, fir, sedan-car, sports-car, sailing-boat, pedal-boat. All items are well known to Israeli children of the ages included in the study. The number of subjects at the three levels were: 2nd graders -

20, 37, 65; 8th graders - 27, 62, 98; college students - 22, 38, 50.

Procedure. The task was administered in groups. Each subject was handed a sheet of paper with the following instructions:

"Imagine that creatures from another world, which are very much like human beings, have landed on Earth. You have to instruct them, through the use of pictures, to identify things (such as a chair, a carrot, a tool). Imagine that you have at your disposal a collection of pictures which can include any picture you wish.

Which pictures would you choose to show them, so that when they encounter an object they will know whether or not it is a _____ ?"

The blank line was completed with one of the 19 items mentioned above.

Results and Discussion

Classification of answers. For the purposes of the present paper each answer list was characterized by its values on two dimensions:

a) Mentioning of multiple positive instances of the category: A list either contained multiple positive instances of the category (MP+) or did not (MP-).

b) Mentioning of negative instances of the category: A list either contained one or more negative instance (N+) or did not contain any negative instance (N-). A list was regarded as N+ if it contained at least one item from a category which shares the same immediate superordinate category with the item in question (for example, for bird "I'll show a picture of a butterfly and cross it out"; for bulldog, "I'll show a picture of a German shepherd").

The answers were independently evaluated by two judges. Agreement was very high (over 98%). The four resulting patterns of answers are discussed below.

a) (MP-/N-): Absence of multiple positive instances and absence of negative instances. This pattern was understood to mean that the subject relied on the DSD to indicate the variability of the category in question.

b) (MP-/N+): Absence of multiple positive instances and presence ofnegative instances. This pattern was understood to mean that the variability of the category in question was smaller than what the subject considered to be the norm, and had to be explicitly delimited by presenting negative instances.

c) (MP+/N-): Presence of multiple positive instances and absence of negative instances. This pattern was understood to mean that the variability of the category in question was larger than what the subject considered to be the norm, and had to be explicilty expanded by presenting more than one positive instance.

d) (MP+/N+): Presence of both multiple positive instances and negative instances. Here nothing can be inferred concerning the variability of the category in question relative to DSD.

Analysis. A corrolary of the claim that there exists a DSD for categories, is that category variability is explicitly indicated when DSD is inappropriate. Since the higher the level of a category the larger its variability, we expected the incidence of the MP+/N- pattern to increase with level. The opposite was expected for the MP-/N+ pattern: We expected it to be more prevalent the lower the category level. The relationship between category level and the incidence of the two patterns is depicted in Figure 1.

Since support for these two predictions is a prerequisite for any further analysis we first tested the main effect of level for the two patterns of responses in question. The analyses revealed a highly significant main effect of level both for the MP+/N- pattern of answer ($F(2,410) = 23.93$ $p < .001$) and for the MP-/N+ pattern of answer ($F(2,410) = 16.00$, $p < .001$)).

These two results establish that subjects performing the task were sensitive to category variability: The greater the variability the more likely they were to use multiple positive instances; the smaller the variability the more likely they were to use negative instances.

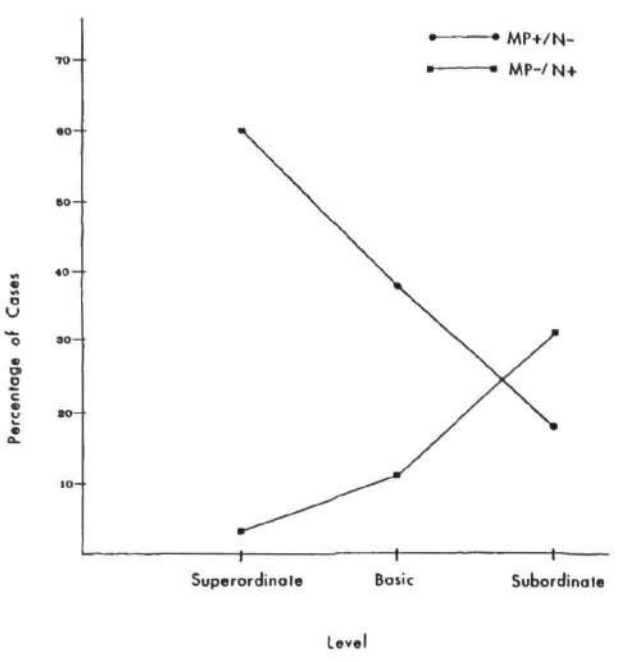


Figure 1: Incidence of MP+/N- and MP-/N+ Patterns of answers at the three levels.

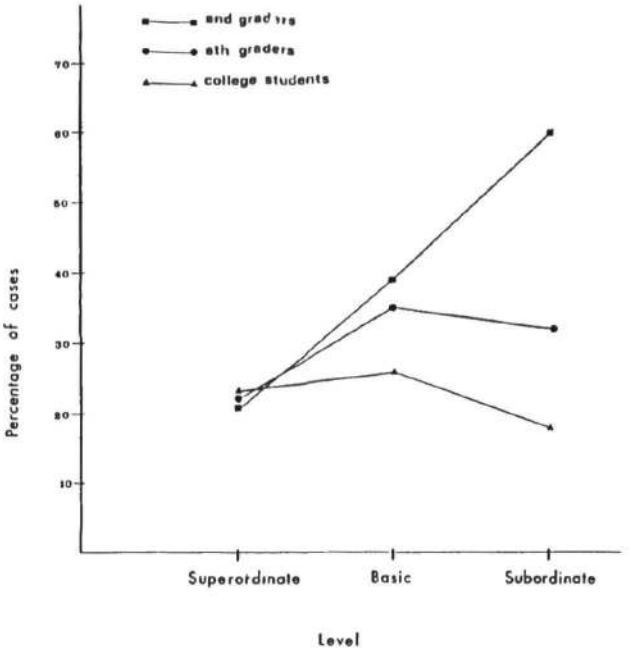


Figure 2: Incidence of the MP-/N- pattern of answers at the three category levels for each age group.

The results for the MP-/N- pattern of responses - the cases where subjects presumably relied on DSD - are presented in Figure 2. A two-way analysis of variance of the responses revealed significant effects of age ($F(2,410) = 3.79$, $p = .023$) and level ($F(2,410) = 3.19, p = .042$), as well as a significant interaction between the two variables ($F(4,410) = 2.42$, $p = .048$). The MP-/N- pattern was more prevalent the younger the subjects (its incidence was .39, .30, and .22 for the three age groups) and the lower the level (an incidence rate of .37, .33, and .22 for subordinate, basic-level and superordinate categories, respectively). Most interesting, the MP-/N- pattern was very common among 2nd graders teaching items from subordinate categories (60%), whereas for older subjects the mode was at the basic-level categories. This trend towards an increase in the size of DSD with age is also evidenced in the finding that for 8th graders the second most frequent MP-/N- cell was that of subordinate categories, while for the college students it was that of superordinate categories.

Our claim that people have a default similarity distance which they use and expect others to use when constructing a new category, provides a coherent and succinct explanation of the results. For each age group there was some category level for which a large proportion of the subjects found it unnecessary to indicate its variability. At the same time, subjects used multiple positive instances to indicate the greater than normal variability of more general categories and negative instances to indicate the relative narrowness of lower-level categories.

The results strongly imply that the size of DSD changes with age: For younger children the default variability was close to that of subordinate categories while for older children and adults the value was closer to the variability of basic-level categories. The evidence of the smaller size of children's default distance for categories is in line with findings of developmental studies of free categorization which indicate that younger children tend to create narrower categories than older ones (Nelson & Bonvillian, 1973, 1978; Saltz, Soller & Sigel, 1972).

The relationship between age and DSD has some implications for models of machine learning. Every model has to allow some similarily distance between items it classifies into the same category. Our findings imply that a model of human learning should initially use a steep generalization gradient and relax this requirement as more knowledge about the world accummulates. Thus, the present findings provide researchers in the field of machine learning with some idea of the size of the default similarity threshold to be installed for grouping stimuli into categories and the changes it should undergo with increased experience.

Bibliography

Fried, L. S. & Holyoak, K. J. (1984). Induction of category distributions: A framework for classification learning. Journal of Experimental Psychology: Learning, Memory and Cognition, 10, 234-257.

Hayes-Roth B. & Hayes-Roth, F. (1977). Concept learning and the recognition and classification of exemplars. Journal of Verbal Learning and Verbal Behavior, 16, 321-338.

Medin, D. L. & Barsalou, L. W. (1987). Categorization processes and categorical perception. In S. Harnad (Ed.) Categorical Perception. Cambridge, England: Cambridge University Press.

Medin, D. L. & Shaffer, M. M. (1978). A context theory of classification learning. Psychological Review, 85, 207-238.

Nelson K. E. & Bonvillian J. D. (1973). Concepts and words in the 18-month-old: Acquiring concept names under controlled conditions. Cognition, 2, 435-450.

Nelson K. E. & Bonvillian J. D. (1978). Early language development: Conceptual growth and related processes between 2 and 4.5 years of age. In K. E. Nelson (Ed.) Children's language (Vol. 1). New York: Gardner.

Rosch, E. (1973). On the internal structure of perceptual and semantic categories. In T. E. Moore (Ed.) Cognitive development and the acquisition of language. New York: Academic Press.

Rosch, E. & Mervis, C. B. (1975). Family resemblances: Studies in the internal structure of categories. Cognitive Development, 7, 573-605.

Saltz E., Soller, E. & Sigel I. E. (1972). The development of natural language concepts. Child Development, 43, 1191-1202.

Judgment, Graininess and Categories

Ilan Yaniv and Dean Foster
University of Chicago

Judgment plays a role in reasoning, decision making, and planning. An appraisal of a house's value may influence the seller's consideration of various bids, an estimate of a distance may influence a tourist's decision to walk or ride a bus, and the estimated arrival time of a guest may change the dinner schedule. Clearly optimal choice in such cases may hinge on having good judgmental estimates (Tversky & Kahneman, 1974; Dawes, Faust, & Meehl, 1989).

How good a judgmental estimate is, depends on how close it is to the "truth". However, this is not the whole story. People often communicate the "fineness of grain" of their judgmental estimates. For example, compare the following:

(a) John has promised to be at a meeting at "5:00 pm". John arrives at 5:15 pm.

(b) Bill has promised to be at a meeting around "fivish". Bill arrives at 5:15 pm.

Each of the estimates, 5:00 and fivish, conveys different graininess ("grain size"). The first estimate is fine-grain, whereas, the second is coarse-grain. Although the difference between the promised and actual arrival times is the same in both cases, John's late arrival appears to have been more significant because he has committed himself to greater precision by giving a fine-grain estimate. Intuitively, it seems from this example that the quality of a judgmental estimate depends on its graininess.

Communication of graininess in judgment seems ubiquitous. People have at their disposal alternative, parallel scales differing in graininess that allow them to pack efficiently grain size information in their estimates. For example, an hour and 60 minutes refer to identical durations, but differ in their graininess, thus people might pick one or the other depending on their intended precision. Whereas an hour represents a broad category (potentially including values such as 55, 59, and 64 min),

the estimate 60 minutes conveys greater precision. Other pairs such as three weeks vs 21 days and 1/2 a year vs 183 days, to list but a few, involve similar contrasts.

What is the role of graininess in judgment? To begin with, we suggest that the psychological impact of a judgmental error depends on the ratio between the size of the error and the graininess of the judgment. This ratio is called grained error (for definitions see Yaniv & Foster, 1990). To illustrate this concept, consider two hypothetical opinions concerning the date the University of Chicago was founded: (a) Bill's opinion is "in the 1880s" (b) John's opinion is "1885". In light of the correct answer (1892), both opinions are wrong. But the graininess of Bill's estimate is a decade thus his estimate is about "one grain away" from the truth, whereas the graininess of John's estimate is one year, thus his estimate is several (more than one) grains away from the truth.

How do people select the appropriate "grain size" for their judgmental estimates? What role does graininess play in evaluating the accuracy of judgments made by other people? We briefly report here the results of two representative studies (see Yaniv & Foster for full report). In the following section, we outline the paradigm of the first study, our hypotheses and the results.

Grain-Scale study

Method. We asked a group of 44 University of Chicago students to estimate a variety of quantities, such as, "The US population in 1987," "Air distance between Chicago and New York," and " Number of American symphony orchestras." The sample question shown in Figure 1, "The date the University of Chicago was founded," illustrates the format of the questionnaire. Various scales were provided in increasing order of fineness of grain. The top scale was provided as an option in case the respondents didn't know anything about the subject of the question in which case they were supposed circle the whole range. However, if they felt they knew more, they could select finer grain scales. They could circle an interval representing a century on

the second scale (grain size = a century), or else, if they felt they knew the answer with more precision, they could circle an interval representing a fifty-year period on the third scale, etc. They were also given the option of writing down the exact year in the space provided above the sixth scale (grain size = 1 year). Thus, they were supposed to make only one grain-scale estimate per question by selecting an appropriate scale and an interval on that scale. This method jointly elicited their best guesses (approximately the midpoint of the selected interval) along with the graininess of their judgments (represented by the width of this interval).

As part of the analysis, we calculated the grained error of each estimate. For illustration, the grained error of the answer shown in Figure 1 is +2 because it is 2 units away from the interval containing the correct date. (In contrast, the grained error of the estimate 1900-1909 would have been -1, while the grained error of 1890-1899 would have been 0 because this interval contains the correct date.)

Hypotheses. One hypothesis is that in selecting grain sizes for their judgments, individuals signal to others the magnitude of judgmental error that they expect. Thus, individuals estimate historical dates in decades if they believe their best guess might be (on the average) 10 years off the truth, and they give estimates in centuries if they expect an average (absolute) error on the order of 100 years. A corollary of this hypothesis is that the mean absolute grained error across judgments should average one.

Another possible hypothesis is that grain size of a judgment is like a "confidence interval" -- an interval that includes the true answer with measured certainty, for example, with a probability of 95% or 99%. If this is true, then grain-scale judgments should include the correct answers with a high probability.

Results. The distribution of the grained errors across all grain-scale judgments is plotted in Figure 2. The most striking result is that only 46% of the grain-scale judgments actually contained the correct answer; 75% of the judgments had a grained error that was either -1, 0, or +1; and 95% of the judgments had a grained error that

ranged from -5 to +5. Thus, grain-scale judgments do not appear to "behave" like confidence intervals that contain the correct answers with a high probability. These results are consistent with Alpert and Raiffa's (1982) who found that judgmental confidence intervals tend to be too narrow (see also Yates, 1990).

It is interesting that grain-scale judgments tend to be "too finely-grained" and hence exclude the truth so frequently. Clearly, giving coarsely-grained estimates can reduce grained error and increase the chances that the truth is included in the estimate. For instance, a coarse-grain estimate for the date the University of Chicago was founded "1700-1900" will generate a lower grained error than a fine-grain estimate such as "in the 1880s."

Grained Error vs Graininess: A Tradeoff

In everyday situations, however, giving excessively coarse judgments is discouraged by linguistic/social norms which imply that speakers should be truthful and appropriately informative (Grice, 1975). Judgmental estimates are expected to be truthful, that is, have low grained error. But, they are also expected to be informative, or finely-grained. For instance, the forecast that inflation rate in the US in 1990 will be between 1% and 55% appears vacuous although it is quite likely to be truthful. Consider for example two estimates of the age of a particular person: (a) "25 to 40 years" (b) "40 to 42 years." Suppose that the person's true age is 39 years. In retrospect, which estimate would we prefer to have had? The first estimate is coarser than the second and has a lower grained error. The second estimate is more informative, but it has a higher grained error. The evaluation of estimates appears to involve a tradeoff between grained error and graininess. A model which formalizes this tradeoff between the truthfulness (grained error) and informativeness (graininess) of a given judgment is presented in Yaniv and Foster (1990).

Evaluation Study

We briefly outline a study designed to test our model. We told our respondents to "imagine that you are a senator preparing an argument. You solicit quick judgmental estimates for some missing information from two of your aides. Later, you compare these estimates to the truth." We presented to them pairs of grain-scale estimates of the target questions along with the correct answer. For example:

> Amount of money spent on education by the U.S. federal government in 1987?
> _____ Aide A responds: $20 to 40 billion
> _____ Aide B responds: $18 to 20 billion
> The actual answer was: $22.5 billion. Which aide is more credible?

In each case, respondents were supposed to compare the estimates in terms of their quality, specifically, to indicate which of the two aides they would prefer to consult in the future.

Results. Our model provided a weighted measure of the "quality" of each estimate based on its grained error and graininess (width of the judgmental interval). We found that the preferences implied by our model predicted the preferences indicated by our respondents.

Final Comments

We have discussed two major ideas. First, judges seem to select grain sizes which estimate the expected error of their estimates. Second, the graininess of a given judgment depends on the tradeoff between the judge's pragmatic (and conflicting) needs to be truthful and informative at the same time (for further discussion see Yaniv & Foster).

It would be interesting to speculate on the representation that may give rise to grain-scale judgments. It is conceivable that grain-scale judgments rely on permanent hierarchical memory representations similar to those underlying natural language categories (e.g., Smith & Medin, 1981). Recent work (Huttenlocher et al., 1990)

suggests that the dates of particular autobiographical events (e.g., the date we saw a particular movie) might be coded at multiple levels (e.g., "January 20", "January" and "Winter quarter".) It is possible that numerical information is generally coded in such structures. For example, historical events (e.g., the date the University of Chicago was founded) may be coded at multiple levels varying in fineness (similar to the levels represented in Figure 1) and with different degrees of certainty. Thus we may remember the target event was "definitely a 19th century event", "most likely in second half of the 19th century," "possibly in the 1880s," etc. Grain-scale judgments may result from the confluence of several such sources of information.

References

Alpert, M., & Raiffa, H. (1982). A progress report on the training of probability assessors. In D. Kahneman, P. Slovic, & A. Tversky, Judgment under uncertainty: Heuristics and biases. New York: Cambridge University Press.

Dawes, R.M., Faust, F., & Meehl, P.E. (1989). Clinical versus actuarial judgment. Science, 243, 1668-1674.

Grice, H.P. (1975). Logic and conversation. In P. Cole & J.L. Morgan (Eds.), Syntax and semantics: Vol. 3. Speech acts (pp.64-74). New York: Academic Press.

Huttenlocher, J., Hedges, L., & Bradburn, N. (1990). Reports of elapsed time: Bounding and rounding processes in estimation. Journal of Experimental Psychology: Learning, Memory, and Cognition, 16, 196-213.

Smith, E.E., & Medin, D.L. (1981). Categories and concepts. Cambridge, MA: Harvard University Press.

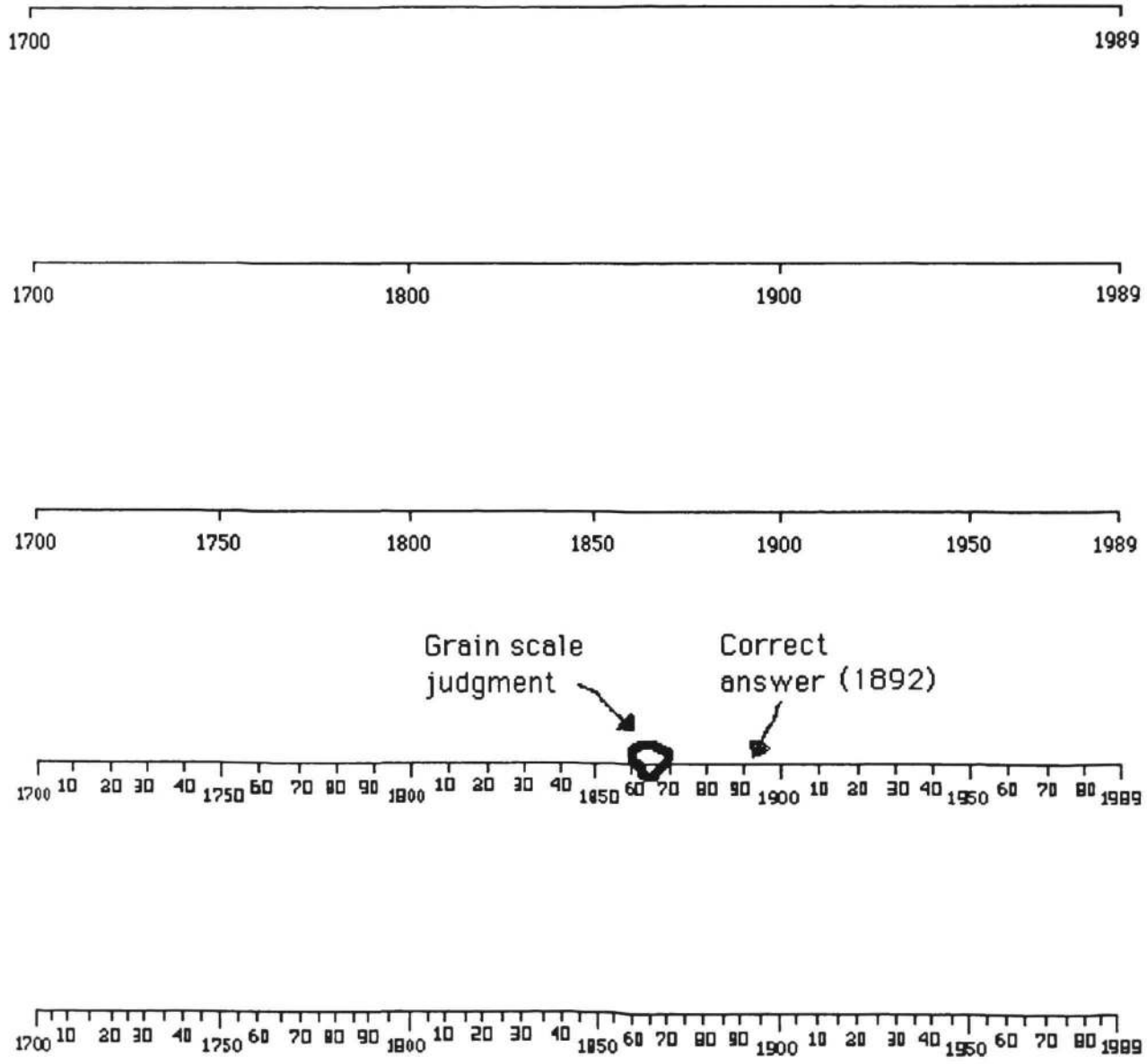
Tversky, A., & Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. Science, 185, 1124-1131.

Yaniv, I., & Foster, D.P. (1990). Graininess of judgment. Center for Decision Research, University of Chicago.

Yates, J.F. (1990). Judgment and Decision Making (pp. 75-111). Englewood Cliffs, NJ: Prentice Hall.

Figure 1.

Date the University of Chicago founded?

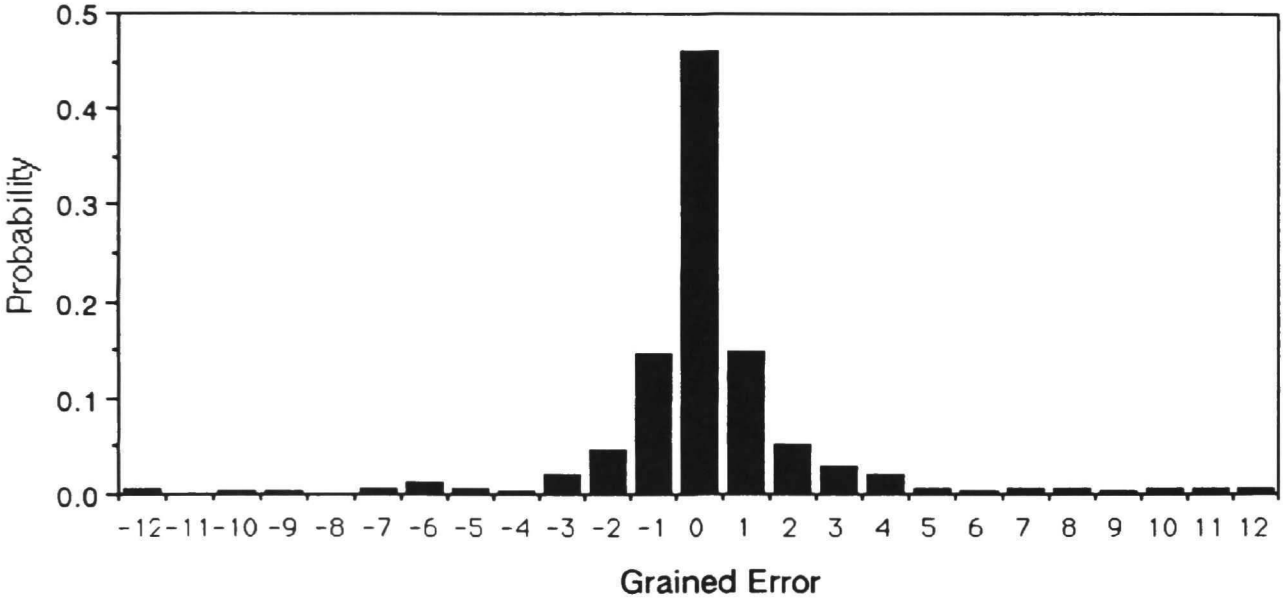


The exact year is: _____



Figure 2.

Grain-Scale study: Distribution of Grained Errors



Learning Attribute Relevance in Context in Instance-Based Learning Algorithms

David W. Aha
Dept. of Information & Computer Science
University of California, Irvine
Irvine, CA 92717
aha@ics.uci.edu

Robert L. Goldstone
Department of Psychology
University of Michigan
Ann Arbor, MI 48109
rob_goldstone@ub.cc.umich.edu

Abstract

There has been an upsurge of interest, in both artificial intelligence and cognitive psychology, in *exemplar-based* process models of categorization, which preserve specific instances instead of maintaining abstractions derived from them. Recent exemplar-based models provided accurate fits for subject results in a variety of experiments because, in accordance with Shepard's (1987) observations, they define similarity to degrade exponentially with the distance between instances in psychological space. Although several researchers have shown that an attribute's relevance in similarity calculations varies according to its *context* (i.e., the values of the other attributes in the instance and the target concept), previous exemplar models define attribute relevance to be invariant across all instances. This paper introduces the GCM-ISW model, an extension of Nosofsky's GCM model that uses *context-specific* attribute weights for categorization tasks. Since several researchers have reported that humans make context-sensitive classification decisions, our model will fit subject data more accurately when attribute relevance is context-sensitive. We also introduce a process component for GCM-ISW and show that its learning rate is significantly faster than the rates of previous exemplar-based process models when attribute relevance varies among instances. GCM-ISW is both computationally more efficient and more psychologically plausible than previous exemplar-based models.

1. Introduction

Several studies have shown that the Context Model (Medin & Schaffer, 1978) and the Generalized Context Model (GCM) (Nosofsky, 1986; 1987), two exemplar-based models of categorization, provide excellent fits for subject data from a wide variety of experiments. These models remove the assumption that attributes have equal relevance in similarity computations by introducing a parameter (attribute weight) for each attribute, whose value is determined by some exterior attention mechanism. Humans are hypothesized to selectively attend to attributes to optimize their classification behavior (Nosofsky, 1986). These models differ from previous exemplar models (e.g., Reed, 1972) in that they define similarity to decrease exponentially with psychological distance, in accordance with Shepard's (1987) numerous empirical observations on stimulus generalization. However, these models ignore evidence that attribute relevance varies depending on the context of the classification task (Tversky, 1977; Barsalou, 1982; Roth & Shoben, 1983; Medin & Edelson, 1988). Therefore, they cannot be expected to provide accurate fits when attribute relevance varies according to context, which occurs frequently in real-world classification tasks.

For example, consider the problem of predicting whether a pro-life politician will endorse proposed legislation on abortion rights. As with most real-world categorization tasks, some attributes should be given more attention than others. In this case, dimensions such as

"past voting record" should be weighted more than dimensions such as "height." Relative attribute relevance differs depending upon the prediction task (Aha & McNulty, 1989; Aha, 1989) (i.e., "past voting record" is far less relevant than "height" when predicting the ability to dunk a basketball). However, an attribute's relevance to a categorization task often also depends on its *context* – the values of the other attributes in an instance. For example, the relevance of the "past voting record" attribute will be low if the "percentage of pro-choice constituency" attribute has a high value (due to pressure from pro-choice political action groups). However, it will be high if the "seek re-election" attribute value is "false", which diminishes the influence of political action groups. Context sensitive attribute weights are required to derive an appropriate psychological space and satisfy the attention-optimization hypothesis when attribute relevance is context-dependent.

In this paper, we introduce the *GCM-ISW* (**I**nstance-**S**pecific **W**eights) model, an extension of the GCM that adds a set of attribute weight parameters for each instance for each target concept. Section 2 describes evidence that the GCM-ISW is more *computationally efficient* (i.e., records significantly faster learning rates) than previous exemplar-based process models when attribute relevance is context-dependent. Several researchers have reported evidence that humans make context-sensitive classification decisions when attribute relevance is dependent on context. In Section 3, we review this evidence and discuss alternative weighting schemes for exemplar-based models.

2. Instance-Based Learning Algorithms

This section describes a sequence of four, comprehensive *instance-based learning* (IBL) algorithms, which are exemplar-based process models. The attribute weights in the first (and simplest) model, named *GCM-NW* (**N**o **W**eights), are fixed to be equal. We describe evidence that a process model for the GCM, named *GCM-SW* (**S**ingle set of attribute **W**eights), learns significantly faster than the GCM-NW when the relevance to classification judgements varies among attributes. The third model, named *GCM-MW* (**M**ultiple sets of attribute **W**eights), employs a separate set of attribute weights per target concept and learns significantly faster than the GCM-SW when attribute relevance varies among target concepts. The final model, GCM-ISW, employs a separate set of attribute weights for each instance for each target concept. We present evidence that it learns significantly faster than GCM-MW when attribute relevance varies among instances.

2.1 GCM-SW: Learning Attribute Relevance

IBL algorithms input a sequence of training instances, drawn from an n-dimensional *instance space*, where n is the number of attributes used to describe each instance. A subset $p < n$ of these attributes, called *predictors*, are used to predict values for the remaining $(n - p)$ *targets*. In this paper, we assume that predictors have numeric values and target attributes have binary values: "positive" and "negative."[1] Positive target values are understood to be members of the *target concept*. For each target, IBL algorithms yield one *concept description* which contains the processed training instances and a set of attribute weight settings. Given an instance x and its similarity with each instance in target a's concept description, IBL algorithms can predict whether x is a member of concept a.

[1] Stanfill and Waltz (1986) describe an interesting IBL algorithm for symbolic-valued predictors. Kibler, Aha, and Albert (1989) address the issue of numeric-valued targets.

IBL algorithms use a *similarity function*, defined over the predictor attributes, to compute the similarity of the instance to be classified with the previously processed training instances. GCM-SW's similarity function is $\text{similarity}(x, y) = e^{-\text{distance}(x,y)}$, where

$$\text{distance}(x, y) = s\sqrt{\sum_{i=1}^{p} w_i(x_i - y_i)^2},$$

where parameter s (set to 10 in all our simulations) is GCM-SW's parameter that determines the slope of the exponential decay and w_i is GCM-SW's weight for attribute i. Values for attribute weights are always initialized to $\frac{1}{p}$, range in $[0, 1]$, and are normalized to sum to 1.

Given these similarities, a *memory updating function* modifies the attribute weights for the instances in a's concept description and, afterwards, always adds x to a's description.[2] The GCM-SW training algorithm is: (where cd_a is target a's concept description)

1. $\text{cd}_a \leftarrow \emptyset$
2. FOR EACH $x \in$ training set DO
   2.1 FOR EACH $y \in \text{cd}_a$: compute similarity(x, y)
   2.2 FOR EACH $y \in \text{cd}_a$: FOR EACH predictor attribute i: adjust_weight(i, x, y, a)
   2.3 $\text{cd}_a \leftarrow \text{cd}_a \cup \{x\}$

Attribute weights denote the estimated relevance of an attribute for a categorization task. Each predictor i's weight is computed using a function of the estimated conditional probability that two instances will have the same class, given that their similarity is high and the difference of their values for i is small. If we denote this probability at time t as $\text{Pr}_i(t)$, then the attribute weight for i after t training instances have been processed is $\text{Pr}_i(t) - (1 - \text{Pr}_i(t))$. Adjust_weight updates estimates of conditional probability as follows:

$$\text{Pr}_i(t+1) = \text{Pr}_i(t) + (r - \text{Pr}_i(t)) \times \text{similarity}(x, y) \times e^{-s|x_i - y_i|} \times \rho,$$

where Boolean variable r is 1 only if x_a equals y_a and ρ is a learning rate parameter, which is set to 0.01 for GCM-SW and GCM-MW in our simulations. The size of the update to i's conditional probability increases exponentially with linear decreases in both distance(x, y) and $|x_i - y_i|$. Therefore, attribute i's weight is most strongly influenced by highly similar instances with similar values for i.

The classification accuracy of our IBL algorithms is measured using a *classification function*, which inputs the computed similarities for target a and generates a class prediction (i.e., "positive" or "negative"). The probability that instance x will be a member of concept a is estimated as follows:

$$\text{Pr}(x_a = \text{"positive"}) = \frac{\sum_{y \in \text{cd}_a} \text{similarity}(x, y) \times (y_a = \text{"positive"})}{\sum_{y \in \text{cd}_a} \text{similarity}(x, y)}$$

Instance x is predicted to be a member of concept a only if this value is above 0.5. All the IBL algorithms in this paper use the following testing algorithm (for each target attribute

[2] Aha, Kibler, and Albert (in press) analyze IBL algorithms that significantly reduce storage requirements.

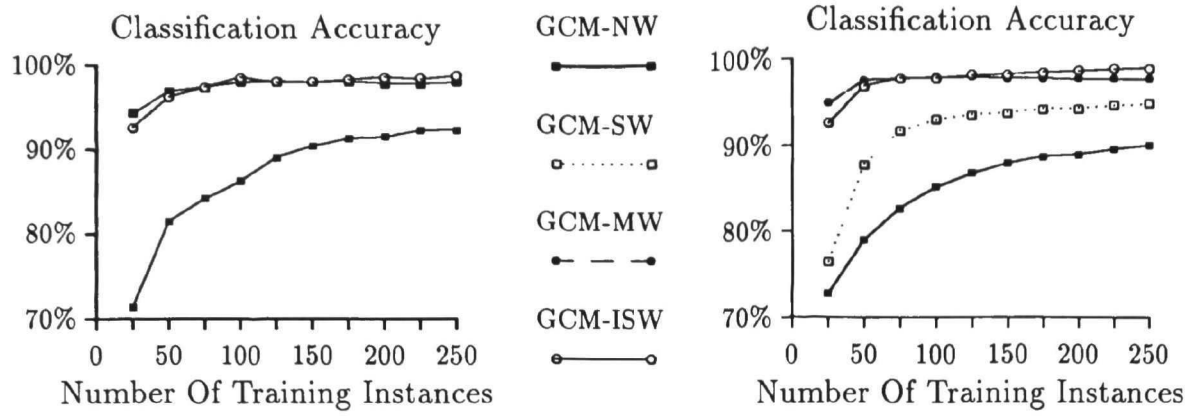


Figure 1: Learning curves for the four IBL algorithms. Left: GCM-NW learns slowly when attributes have different relevance. GCM-SW and GCM-MW behave identically in this simulation since there is only one target concept. Right: GCM-SW learns slowly when each attribute's relevance differs among target concepts. All our curves are averaged over 20 pairs of training and test sets with 250 and 100 instances respectively. Values for predictor attributes are selected randomly from [0, 1] according to a uniform distribution.

a): (1) compute the current training instance x's similarity to the instances in a's concept description, (2) compute the probability that x_a is "positive", and (3) output "positive" for this classification if this probability is above 0.5 (otherwise, output "negative").

GCM-SW's attribute weights are useful when attribute relevance varies among predictors. To show this, we compared its performance with the performance of GCM-NW, whose weights remain fixed with value $\frac{1}{p}$. GCM-NW learns slowly when attribute relevance differs among the predictors. The graph in the left of Figure 1 shows the average learning curves for a simulation with one target concept and ten predictors, only one of which was relevant. Target concept members were defined to be those whose relevant attribute's value was greater than 0.5. As expected, GCM-SW's average accuracy (measured across the ten applications to the test set per trial) is significantly greater than GCM-NW's ($t(19) = 4.54, p < 0.001$). However, since the GCM-SW model uses the same setting of attribute weights for all targets, it performs relatively poorly when the relative relevance of attributes differs greatly among target concepts (Aha & McNulty, 1989) or when relative attribute relevance varies among instances. The right-hand graph in Figure 1 shows the average learning curves when the artificial domain is extended to contain an additional three target concepts, where each of the four target concepts have a single (different) relevant predictor. GCM-SW's learning curve rises slowly because it is unable to learn concept-dependent attribute relevances: its weights for the four relevant attributes each converge to 0.25. GCM-SW's average classification accuracy is significantly lower than GCM-MW's ($t(19) = 5.33, p < 0.001$).

2.2 GCM-MW: Learning Concept-Dependent Attribute Relevance

GCM-MW's concept-dependent similarity function is $\text{similarity}(a, x, y) = e^{-\text{distance}(a,x,y)}$, where w_{a_i} denotes the weight of attribute i for target concept a in

$$\text{distance}(a, x, y) = s\sqrt{\sum_{i=1}^{p} w_{a_i}(x_i - y_i)^2}.$$

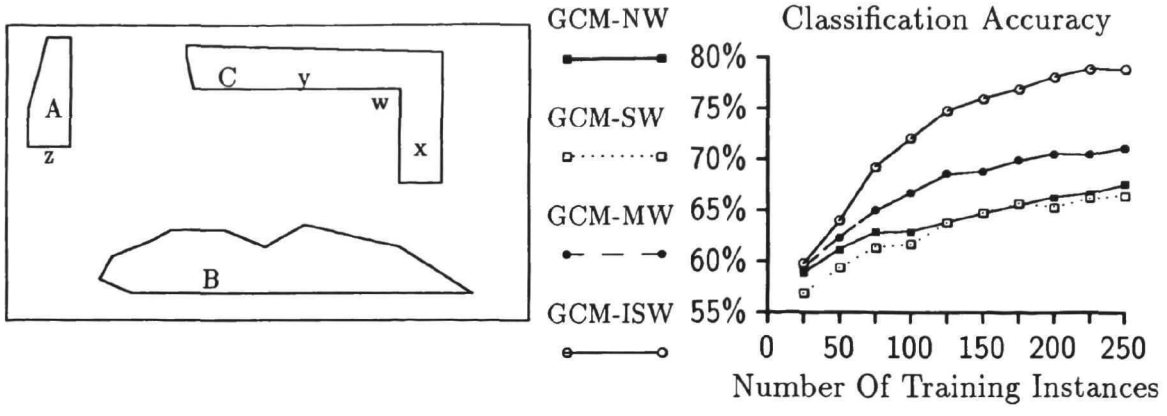


Figure 2: Left: Attribute relevance can vary among instances. Right: GCM-ISW works best in such situations.

The GCM-MW model will outperform the GCM-SW model when attribute relevance varies among target concepts. However, GCM-MW's assumption that an attribute's relevance is invariant across all instances is easily violated. For example, attribute relevance can vary among a concept's disjuncts. Furthermore, it can also vary *within* a disjunct. Figure 2 displays a two-dimensional domain containing three disjuncts of a single target concept. The horizontal attribute is more relevant than the vertical for disjunct A: small perturbations in the horizontal's values will more frequently change disjunct membership status than will perturbations in the vertical's values. The vertical attribute is more relevant for B while C's attributes are approximately equally relevant. However, attribute relevance differs greatly among instances. For example, although both attributes are relevant for classifications made by instance w, the horizontal attribute is more relevant for x and less relevant for y. Finally, z's vertical attribute is more relevant. GCM-MW's learning rate can be significantly reduced when attribute relevance varies among instances. Figure 2 displays the average learning curves when the learning task is changed so that each of the four concepts is defined by a set of five disjuncts. In this case, each disjunct is defined by a single relevant attribute and each attribute in the domain is relevant to exactly two disjuncts overall. The threshold values for inclusion in a disjunct were below 0.14 for the disjuncts of the first two target concepts and above 0.86 for the latter two target concepts. GCM-MW's average accuracy is significantly lower than GCM-ISW's ($t(19) = 3.85, p < 0.002$).

2.3 Learning Context-Sensitive Attribute Relevance

GCM-ISW differs from GCM-MW in that it learns *instance-specific* attribute weights, one for each ⟨attribute,instance,target⟩ triplet. This provides greater flexibility than found in GCM-MW: GCM-ISW removes the assumption that attribute relevance is invariant among a target concept's saved instances.

GCM-ISW's instance-specific weights can be easily misapplied. For example, if the only relevant attribute for instance z in Figure 2 is the vertical attribute, then z will appear to be very similar to x, which is located far from z in this instance space. Therefore, instance-specific weights should be used only when the instance being classified is highly similar to the classifying instance. GCM-ISW solves this problem by learning both concept-dependent weights

(as is done in GCM-MW) and a separate set of instance-specific weights. (Adjust_weight updates each saved instance's attribute weights when classifying each subsequently presented training instance.) GCM-ISW's similarity function then combines these two sets of weights to compute the *context*-specific similarity of two instances as follows:

$$\text{distance}(a, x, y) = s\sqrt{\sum_{i=1}^{p} \text{combine_weights}(a, x, y, i) \times (x_i - y_i)^2}.$$

When computing the similarity of a new instance x to previously processed instance y, combine_weights calculates attribute i's *context-specific* weight as follows:

$$\text{combine_weights}(a, x, y, i) = (w_{a_i}(y) \times \text{scale_factor}) + (w_{a_i} \times (1 - \text{scale_factor})),$$

where scale_factor $= (1 - |x_i - y_i|)^c$, $w_{a_i}(y)$ is i's attribute weight for saved instance y, and c is a combination parameter that determines the relative impact of the concept-dependent and instance-specific attribute weights in calculating the context-sensitive weight.[3] Combine_weights uses instance-specific weights more confidently when the difference of the values for i is small. This reduces the frequency with which instance-specific weights are used when the distance between instances is large. After GCM-ISW computes similarities, it updates the conditional probabilities and attribute weights for both its concept-dependent and its instance-specific weights. GCM-ISW performed significantly better than the other models in the third simulation and performed as well as GCM-SW and GCM-MW in the first and second simulations respectively.

3. Discussion: Supporting Evidence and Alternative Models

While formal psychological models involving context-specific weight learning do not exist, there is a plethora of psychological data suggesting the existence of such specific weighting systems. Nosofsky's (1986) GCM model treats attribute weights as parameters that can be assigned experimentally-derived values to accurately fit subject data. However, more flexible weighting systems, namely those that employ context-sensitive weights, are required to accurately fit subject data and increase learning rate when attribute relevance varies among instances. These weights can also be used to decrease storage requirements: attributes with low relevance can be discarded without sacrificing classification accuracy (Smith & Medin, 1981). Aha (1989) described simulations of IBL algorithms that drop both attributes *and* instances and, simultaneously, increase learning rates for real-world classification tasks.

Many researchers agree that models of categorization should be context-sensitive. Roth and Shoben (1983) and Barsalou (1982) argue that an instance's context influences its perceived typicality and determines which of its attributes receives attention. For example, Barsalou noted that, while some attributes of "basketball" (e.g., "round") are always salient, others (e.g., "floats") only become salient (i.e., quickly retrieved) in contexts involving water. This provides psychological support for the GCM-ISW model: people on a luxury liner attend more to the "floats" attribute when the ship is sinking (to judge whether objects are members of the "can support me in the water" category) than when it is in port. Goldstone, Medin,

[3] We used $c = 0.5$ for our simulations. We also set GCM-ISW's learning rate parameter to be higher (0.1) when it updates instance-specific weights. This is needed because, given any one training instance, few other training instances are highly similar to it. However, when updating concept-dependent weights, there will be several highly similar pairs of instances.

and Gentner (in press) argue that, when comparing instances, the influence that one attribute has depends on the other attributes that are shared by the instances. Medin and Edelson (1988) also suggested using context-specific attribute weights. When an instance is correctly classified, their proposed process model assigns high relative weights to the attributes shared by the classifying instance and the instance being classified. Misclassifications result in assigning higher weights to attributes that are *not* shared by these two instances.

Several alternative weighting schemes have been proposed for exemplar-based process models. Nosofsky, Clark, and Shin (1989) considered *value-specific* weighting algorithms. However, these are not as flexible as instance-specific weighting algorithms: value-specific weights for some attribute i will not work well when i's relevance varies over instances that have the same value for i. In another example, Medin and Shoben (1988) present examples that suggest an *instance-directed* attribute-weighting scheme, whereby the influence of one attribute depends on the other attributes that are present. For example, while "White" is more similar to "Gray" than is "Black" for the attribute "hair," exactly the opposite pattern emerges with the attribute "clouds." This suggests extending the instance-specific weighting method to distinguish between directions along attribute dimensions. For instances of hair, the gray-black distance is widened while the gray-white distance is reduced. In any case, a single predefined weight for the "color" dimension will not survive changes of context.

The GCM-ISW model adds an enormous number of parameters into the GCM model. Although GCM-ISW increases learning rate, its additional parameters are not needed when attribute relevance remains constant across the entire dimension. We are currently developing a more elaborate IBL algorithm that can learn which parameters should be permanently fixed without need for subsequent attention. The algorithm would initially assume that all dimensions are weighted equally for all categories. If this assumption does not yield sufficiently fast learning rates, then the system would relax its assumptions and allow an attribute's weight to vary across categories. The assumption that weights are fixed across instances could also be automatically relaxed. Shifts in the target concept description could lead to more or less specific weighting algorithms in attempts to maximize classification accuracy while minimizing the number of unique weights that are postulated.

IBL algorithms that learn context-specific attribute weights resemble rule-based learning algorithms. By weighting dimensions selectively on the basis of their category diagnosticity, the instance-based systems are qualitatively distinguished from the simple storage of instances in a "raw form." Although instance information is not discarded, it is selectively emphasized. This representation is similar to that used for rules. For example, consider the concept of legal-sized suitcases (i.e., those with lengths less than five feet). An instance-directed weighting algorithm could learn a high weight for $4'\,9''$ in the positive direction and a low weight in the negative direction for legal-sized suitcases. This is similar to the rule "if $4'\,9''$ or less, then legal-sized luggage, otherwise illegal."

4. Conclusion

Results from simulations suggest that previous exemplar models that selectively weight attribute dimensions, while better than no selective weighting at all, can be improved by representing context-sensitive attribute weights. We introduced GCM-ISW, an extension of Nosofsky's (1986) GCM model that learns context-sensitive weights by combining concept-

dependent and instance-specific attribute weights. Our results with simulations using a process model for the GCM-ISW show that its learning rate is significantly faster than the learning rates of previous process models for exemplar-based models (Aha & McNulty, 1989). We plan to show that the GCM-ISW model will fit subject data more accurately than will a process model for the GCM when attribute relevance varies among instances.

Acknowledgements

We would like to thank Marc Albert, Dale McNulty, Douglas Medin, and Mike Pazzani for providing comments on an earlier draft of this paper.

References

Aha, D. W. (1989). Incremental, instance-based learning of independent and graded concept descriptions. In *Proceedings of the Sixth International Workshop on Machine Learning* (pp. 387–391). Ithaca, NY: Morgan Kaufmann.

Aha, D. W., Kibler, D., & Albert, M. K. (in press). Instance-based learning algorithms. *Machine Learning.*

Aha, D. W., & McNulty, D. (1989). Learning relative attribute weights for independent, instance-based concept descriptions. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society* (pp. 530–537). Ann Arbor, MI: Lawrence Erlbaum Associates.

Barsalou, L. W. (1982). Context-independent and context-dependent information in concepts. *Memory & Cognition, 10,* 82–106.

Goldstone, R. L., Medin, D. L., & Gentner, D. (in press). Attributes, relations, and the non-independence of features in similarity judgments. *Cognitive Psychology.*

Kibler, D., Aha, D. W., & Albert, M. (1989). Instance-based prediction of real-valued attributes. *Computational Intelligence, 5,* 51–57.

Medin, D. L., & Edelson, S. M. (1988). Problem structure and the use of base rate information from experience. *Journal of Experimental Psychology: General, 117,* 68–85.

Medin, D. L., & Schaffer, M. M. (1978). Context theory of classification learning. *Psychological Review, 85,* 207–238.

Medin, D. L., & Shoben, E. J. (1988). Context and structure in conceptual combination. *Cognitive Psychology, 20,* 158–190.

Nosofsky, R. M. (1986). Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General, 15,* 39–57.

Nosofsky, R. M. (1987). Attention and learning processes in the identification and categorization of integral stimuli. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 13,* 87–108.

Nosofsky, R. M., Clark, S. E., & Shin, H. S. (1989). Rules and exemplars in categorization, identification, and recognition. *Journal of Experimental Psychology: Learning, Memory, & Cognition, 15,* 282–304.

Reed, S. K. (1972). Pattern recognition and categorization. *Cognitive Psychology, 3,* 382–407.

Roth, E. M., & Shoben, E. J. (1983). The effect of context on the structure of categories. *Cognitive Psychology, 15,* 346–378.

Shepard, R. N. (1987). Toward a universal law of generalization for psychological science. *Science. 237,* 1317–1323.

Smith, E. E., & Medin, D. L. (1981). *Categories and concepts.* Cambridge, MA: Harvard University Press.

Stanfill, C., & Waltz, D. (1986). Toward memory-based reasoning. *Communications of the ACM, 29,* 1213–1228.

Tversky, A. (1977). Features of similarity. *Psychological Review, 84,* 327–352.

Effects of Background Knowledge on Family Resemblance Sorting

Woo-Kyoung Ahn
University of Michigan

Abstract

Previous studies on category construction have shown that people have a strong bias of creating categories based only on a single dimension. Ahn and Medin (1989) have developed a two-stage model of category construction to explain why we have categories structured on the basis of overall similarity of members in spite of this bias. The current study investigates effects of background knowledge on category construction. The results showed that people created family resemblance categories more frequently when they had a priori knowledge on prototypes of potential family resemblance categories. It was also found that people created family resemblance categories much more frequently when they had knowledge on underlying dimensions which integrated surface features of examples. How the two-stage model should be extended is discussed.

Introduction

Generally, it has been argued that natural categories lack defining features that are true to all members in the same category (Rosch & Mervis, 1975; Smith & Medin, 1981). Instead, natural categories are considered to have family resemblance (FR) structure: members in the same category are generally similar to each other but members in different categories are dissimilar to each other. Rosch (1975) argued that FR categories might be created because people would try to achieve a compromise between maximizing within-category similarity and minimizing between-category similarity. However, previous studies showed that people rarely sort unclassified examples based on their overall similarity (Ahn, 1990; Ahn & Medin, 1989; Imai & Garner, 1965, 1968; Medin, Wattenmaker, & Hampson, 1987). Subjects in these experiments showed a strong tendency to sort examples based on values on a single dimension (uni-dimensional sorting) to create categories with defining features.

Ahn and Medin (1989) developed a two-stage model to explain why we have FR structure in spite of this strong bias for uni-dimensional sorting. Although the two-stage model has so far been successful in describing when uni-dimensional sorting and when FR sorting will be observed, the model has only been applied to domains where people do not have any background knowledge on examples to be classified. The present work investigates the effects of various types of background information on creation of FR categories. Specifically, this study is concerned with how knowledge about prototypes and underlying dimensions affect people's sorting behavior.

The first part of the paper briefly reviews previous data from free sorting tasks in knowledge-poor domains, followed by the two-stage model's explanations of these data. The second part presents an experiment in which subjects were provided with either prototypes or theories underlying categories. The final section discusses how the two-stage model should be extended to handle the data obtained in the current experiment.

Previous Results

Medin et al.'s experiments (1987) are the first systematic study showing that people rarely created FR categories even when exemplars to be classified were developed around prototypes on the basis of overall similarity. Figure 1 shows the abstract notation of stimuli used in their experiments. In this figure, the first column (E1, E2, ...) indicates each examples and the next four columns indicate each example's values on 4 dimensions (D1, D2, D3, and D4). These exemplars had characteristic features in the resulting FR categories (i.e., features that are generally true to members in the same category but can also appear in contrasting categories). The exemplars are laid out in two categories, each of which indicate one of the potential FR categories.

	D1	D2	D3	D4
E1	0	0	0	0
E2	0	0	0	1
E3	0	0	1	0
E4	0	1	0	0
E5	1	0	0	0

	D1	D2	D3	D4
E6	1	1	1	1
E7	1	1	1	0
E8	1	1	0	1
E9	1	0	1	1
E10	0	1	1	1

Figure 1. Abstract Notation of Stimuli Used in Medin, Wattenmaker, & Hampson(1987)

This set consists of two prototypes (E1 and E6) and four distortions of each prototype, which were developed by replacing a value of a prototype with the value of a contrastingg prototype. Subjects in Medin et al.'s experiments were asked to create two categories from these exemplars. According to similarity-based clustering models (see Anderberg, 1973; Massart & Kaufman, 1983, for reviews), and category construction models considering predictability of categories as a critical determinant of category construction (Anderson, in press; Fisher 1989), the two categories that the subjects would create should be FR categories (see Ahn, 1990; Ahn & Medin, 1989 for more details). However, almost all subjects in Medin et al.'s experiments sorted examples based on values on a single dimension. Then why do we have FR categories?

Two-Stage Model

To explain these results, Ahn and Medin (1989) proposed a two-stage model of category construction; the first stage involves sorting examples based on one dimension and the second stage involves assigning remaining examples based on their overall similarity to the initially created categories (see also Ahn, submitted; Ahn, 1990, for more details). Ahn and Medin argued that the reason why hardly any FR sorting was observed in Medin et al.'s experiments was because their examples had characteristic features in the resulting FR categories: after subjects carried out uni-dimensional sorting in the first stage, there existed no remaining examples to be classified in the second stage, resulting in uni-dimensional categories.

The two-stage model argues that FR categories can be created as a by-product of the two stages only when the resulting categories have sufficient features. Take an example of the set used in Ahn and Medin (1989) shown in Figure 2 under the

Sufficient set. The FR categories from this set consisted only of sufficient features (i.e., 0's and 2's). Suppose a task is to create two categories and the first dimension was chosen for the most salient dimension. In the first stage, the model categorizes E1, E2, E3, and E4 into one category and E6, E7, E8, and E9 into another category. In the second stage, one of the remaining examples, E5, is grouped with E1, E2, E3, and E4, and E10 is grouped with E6, E7, E8, and E9, based on their overall similarity. As a result, the final categories have a FR structure. On the other hand, when the resulting FR categories have characteristic features such as in the set used in Medin et al.'s experiments and the Characteristic set in Figure 2, the resulting categories are always uni-dimensional.

In Ahn and Medin's experiments, as predicted by the model, no subjects given the Characteristic set created FR categories whereas more than half of those given the Sufficient set created FR categories.

Sufficient Set		Characteristic Set	
E1 0 0 0 0	E6 2 2 2 2	E1 0 0 0 0	E6 2 2 2 2
E2 0 0 0 1	E7 2 2 2 1	E2 0 0 0 1	E7 2 2 2 0
E3 0 0 1 0	E8 2 2 1 2	E3 0 0 2 0	E8 2 2 1 2
E4 0 1 0 0	E9 2 1 2 2	E4 0 1 0 0	E9 2 0 2 2
E5 1 0 0 0	E10 1 2 2 2	E5 2 0 0 0	E10 1 2 2 2

Figure 2. Abstract notation of Sufficient and Characteristic Set

Sorting in Knowledge-Rich Domains

Although existence of sufficient features is shown to be an important determinant of sorting in knowledge-poor domains, it may not be the only one in knowledge-rich domains. Results from Medin et al.'s Experiments 5 and 6 suggest that knowing interproperty relations is an important determinant of creation of FR categories. They used personality descriptions, prototypes of which are descriptions of either an introvert or an extrovert person. When all four dimensions were developed in such a way that they could be related in terms of a single underlying dimension (i.e., introvert / extrovert dimension), most of the participants created FR categories. However, these experiments relied on the background knowledge that was pre-experimentally obtained by the subjects.

To better control the background knowledge, in the current experiment, knowledge on interproperty relationship is experimentally manipulated by providing participants with information on how dimensions could be related in terms of a deeper theory[1]. Take an example used in the current experiment. There were four dimensions which were descriptions of tribes: The first dimension was whether a tribe wore cotton clothes or leather clothes. The second dimension was whether they had monotheism or polytheism. The third dimension was whether they were ruled by hierarchical leaders or by a single leader. The fourth dimension was whether they

[1]There have been debates on what theories are. Although in this paper I used the term, theory, to refer to underlying dimensions and their relationship with surface features, it may be an inappropriate use of the term.

cremated the dead or buried the dead.

At first sight, these dimensions seem to be just a list of independent features, but they can be easily integrated in terms of an agricultural / nomadic dimension. That is, agricultural tribes wore cotton clothes obtained in their farms, were monotheists because they hardly had any chance to learn other types of religion, had hierarchically organized leaders to control farmers living in their areas, and buried the dead near their farms. Nomadic tribes wore leather clothes obtained from their hunting, were polytheists because they had contacted many types of religions while travelling, had a single leader to make flexible and quick decisions in their changing environments, and cremated the dead because they were always on the move.

The underlying theories can be also used to determine prototypes of categories (or ideals, Barsalou, 1985) of the potential FR categories, each of which consists of characteristic values of each category. Therefore, even if FR sorting is observed when interproperty relationships are known, it can be simply attributed to the advantage of knowing the prototypes of the FR categories. To investigate whether there is an additional advantage of knowing theories besides knowing prototypes, it is important to have a condition in which subjects learn prototypes without knowing about interproperty relations.

Previously, prototypes have been considered as a collection of average or the most frequent values on each dimension used to represent members in the same category. In these types of prototype representations, interproperty relations are not necessarily preserved. In the case of agricultural / nomadic examples, knowing that a prototype of one category have values of "monotheism, cotton clothes, burying the dead, hierarchical leaders" does not indicate how these values are related to each other. Therefore, if subjects who receive only prototype information can be compared to those who receive underlying theories, we can examine whether there is an additional advantage of knowing interproperty relations.

Another rationale for having the prototype condition is as follows. People may learn the most typical examples first and then construct categories around the prototypes. One can argue that in the previous experiments, people might have failed to produce FR categories simply because they could not find prototypes. According to this argument, subjects may be more likely to create FR categories if prototypes are given before free sorting tasks. Therefore, introducing the prototype condition allows us to investigate whether the failure to construct FR categories was simply due to failure to identify prototypes.

Method

Basically, subjects received a set of exemplars and were asked to sort them into two groups of any size. There were two groups of subjects depending on whether they received the Characteristic set or the Sufficient set in Figure 2. Half of the subjects in each group received the set instantiated in pictures of flowers and the other half received the set instantiated in descriptions of tribes.

For the flower stimuli, the four dimensions and the values corresponding to 0 and 2 for each dimension were as follows. The first dimension was color with 0 being bright color and 2 being dark color. The second dimension was whether flowers bloomed at night or during daytime. The third dimension was whether flowers bloomed on trees or on grass. The fourth dimension was whether flowers were

located near water or distant from water. (Parts of these stimuli were taken from Nakamura, 1985.) The subjects who received the flower stimuli were explicitly instructed about the four dimensions and their values. For the tribe stimuli, the four dimensions and the values were described in the earlier section. For both types of stimuli, the value 1 was absence of any value on the dimension. The subjects were told that if a certain dimension was not shown in an example, it meant that the information on the dimension was not available.

Within each stimulus set and within each stimulus type, there were three groups depending on types of background knowledge they received before sorting. The types of background knowledge that the subjects received were either none (Control group), prototypes (Prototype group), or underlying theories (Theory group).

The control group who did not receive any information about prototypes was simply asked to sort ten cards in a way that seemed natural to them. There were 78 subjects in the control group who were evenly distributed across the Characteristic and the Sufficient set conditions and across the two types of stimuli.

The Prototype group received eight non-prototype examples and was asked to sort them into two categories in a way that seemed natural to them. With these instructions, they received two prototypes of the potential FR categories (i.e., 0 0 0 0 and 2 2 2 2 in both the Characteristic and Sufficient set) and were told that these were the most typical exemplars of the two categories that they were to create. There were 80 subjects in the prototype group who were also evenly distributed across the various conditions.

The theory group received only the eight non-prototype examples coupled with theories underlying each category they were to create. For the flower stimuli, they were told that one group of flowers attracted a hypothetical class of birds called "champin" and the other attracted a hypothetical class of bees called "trood." They were also told that the champin birds liked bright color, were active at night, flew high, and laid eggs near water and that the trood bees liked dark color, were active during daytime, hovered low, and laid eggs distant from water. For the tribe stimuli, they were told about the theories on the agricultural / nomadic distinction mentioned in the earlier section. Although they could presumably determine the prototypes of the potential FR categories based on the theories, they did not see the actual prototypes during the experiment. There were 77 subjects in the theory group who were evenly distributed to the various conditions.

Results

The results are summarized in Table 1. Numbers in the table indicate percentages of FR sorting within each group within each set. Responses other than creation of FR categories were mostly uni-dimensional sorting.

Comparison between the Characteristic and the Sufficient sets

As in Ahn and Medin (1989), across all groups and across both kinds of stimuli, the Sufficient set led to more FR sortings than the Characteristic set. Chi-square and Fisher's exact tests indicated all the differences between the Sufficient set and the Characteristic set within each group were significant at $p = .05$ except for the theory group with the tribe stimuli (i.e., the difference between 78.9% and 100%).

	Flower Stimuli		Tribe Stimuli		Total	
Group	Char	Suff	Char	Suff	Char	Suff
Control	0	25	11	68	5	47
Prototype	20	90	45	100	33	95
Theory	55	95	79	100	67	98

Table 1. Summary of Results

Comparison among groups within the Characteristic set

In the previous experiments using materials in knowledge-poor domains (Ahn, 1990; Ahn & Medin, 1989), the subjects never created FR categories from the Characteristic set. In this experiment, when the subjects received prototype information on the Characteristic set, subjects created more FR categories (33%) than the control group who did not receive any background information (5%). In the flower stimuli, the difference between the control group (0%) and the prototype group (20%) who received the Characteristic set was not significant but in the tribe stimuli, the difference between the control group (10.5%) and the prototype group (45%) was significant ($p < .001$).

More subjects who received information on underlying theories created FR categories from the Characteristic set (67%) than those who received either prototype information (33%, $p < .05$) or no information (5%, $p < .05$). Within each stimulus type, these differences were also significant.

Comparison among groups within the Sufficient set

For the Sufficient set, almost all subjects in both the prototype group (95%) and the theory group (98%) produced the FR categories. The differences between these two groups and the control group (47%) was significant ($p < .05$). Separate analyses within each type of stimuli also showed the same kind of results.

Discussion

The results of the present experiment can be summarized as follows.

First, results in knowledge-poor domains (Ahn & Medin, 1989) were replicated by the control group in the current experiment. The control group who did not have any background knowledge rarely produced FR categories when the structure of exemplars had characteristic features in the resulting FR categories but a fair amount of FR categories was produced when they had sufficient features. Therefore, the two-stage model's explanation of FR sorting in knowledge-poor domains was once again supported.

Secondly, even when the potential FR categories had characteristic features, the knowledge on the prototypes of the potential FR categories led to the creation of FR

categories. However, the difference between the prototype group and the control group in producing FR categories from the Characteristic set was significant only when the tribe stimuli were used, and it was not significant when the flower stimuli were used. The reason for this difference between the two types of materials is unclear.

Third, when the examples to be classified had characteristic features in the resulting FR categories, the theory group produced much more FR categories than both the control group and the prototype group. The significant difference between the theory group and the prototype group clearly showed that knowing underlying theories is more than simply knowing prototypes that could be derived from the theories.

These results suggest two problems with the current version of the two-stage model. First, the model does not explain why knowing prototypes in advance help people create FR categories. Considering the subjects' protocols, the prototype group seemed to be able to get over the strong bias of uni-dimensional sorting and instead they seemed to try to maximize matches between members of a category and its prototype. If people learn the most typical examples first, then the two-stage model might not be an appropriate account for category construction process.

Secondly, the advantage of knowing interproperty relationships is not explained by the model in its current form. However, sorting based on an underlying theory can be considered as a type of uni-dimensional sorting and therefore, it is not inconsistent with the spirit of the two-stage model.

The current version of the two-stage model assumes that features used for sorting are only surface features used to describe examples. However, people's strong bias to create highly structured categories (i.e., categories with defining features) would make them rely on background knowledge if it can provide them with defining features of categories. In other words, if background information provide them with a way to create a new dimension which allows them to construct categories that meet the task demand and that have defining features, they would rather create the new dimension than using only the given surface dimensions. Therefore, the two-stage model should be extended to allow creation of new dimensions using background knowledge if the new dimensions are better than the surface features in creating categories with defining features.

Conclusion

Based on the series of free sorting experiments conducted so far, it can be concluded that there are at least three ways to obtain FR structure. First, FR categories can be obtained as a result of two stages in which the first stage involves uni-dimensional sorting and the second stage involves assigning exceptions based on overall similarity. Secondly, knowing prototypes of each category before category construction encourages creation of FR categories. Third, the most effective way of producing FR categories known so far is creating a new dimension which can integrate surface features and carry out uni-dimensional sorting based on this dimension.

Acknowledgement

I would like to thank Doug Medin for his helpful advice in developing the

experiment. Also I thank Rob Goldstone and Doug Medin for their comments on earlier drafts. This work was supported by NSF grant BNS88-12913 given to Doug Medin.

References

Ahn, W. (submitted). A two-stage model of category construction.

Ahn, W. (1990). A two-stage model of category construction. Unpublished doctoral dissertation, University of Illinois, Urbana, IL.

Ahn, W., & Medin, D. L. (1989). A two-stage categorization model of family resemblance sorting, Proceedings of the 11th Annual Conference of The Cognitive Science Society, Ann Arbor, MI, 315-322.

Anderberg, M. R. (1973). Cluster analysis for applications. New York: Academic Press.

Anderson, J. R. (1988). The place of cognitive architectures in a rational analysis. Proceedings of the Tenth Annual Conference of The Cognitive Science Society.

Fisher, D. (1987). Knowledge acquisition via incremental conceptual clustering, Machine Learning, 2, 139-172.

Imai, S., & Garner, W. R. (1965). Discriminability and preference for attributes in free and constrained classification. Journal of Experimental Psychology, 69, 596-608.

Imai, S., & Garner, W. R. (1968). Structure in perceptual classification. Psychonomic Monograph Supplements, 2 (9, Whole No. 2).

Massart, D., & Kaufman, L. (1983). The interpretation of analytical chemical data by the use of cluster analysis. New York: John Wiley & Sons.

Medin, D. L., Wattenmaker, W. D., & Hampson, S. E. (1987). Family resemblance, conceptual cohesiveness, and category construction. Cognitive Psychology, 19, 242-279.

Nakamura, G. V. (1985). Knowledge-based classification of ill-defined categories. Memory & Cognition, 13, 377-384.

Rosch, E. (1975). Universals and cultural specifics. In R. Brislin, S. Bochner, & W. Lonner (Eds.), Cross-cultural perspectives on learning. New York: Halsted Press.

Rosch, E., & Mervis, C. B. (1975). Family resemblance: Studies in the internal structure of categories. Cognitive Psychology, 7, 573-605.

Smith, E. E., & Medin, D. L. (1981). Categories and concepts. Cambridge, MA: Harvard University.

Superordinate and Basic Level Categories in Discourse: Memory and Context

Sandra L. Peters and William J. Rapaport
Department of Computer Science
State University of New York at Buffalo
peters@cs.buffalo.edu, rapaport@cs.buffalo.edu

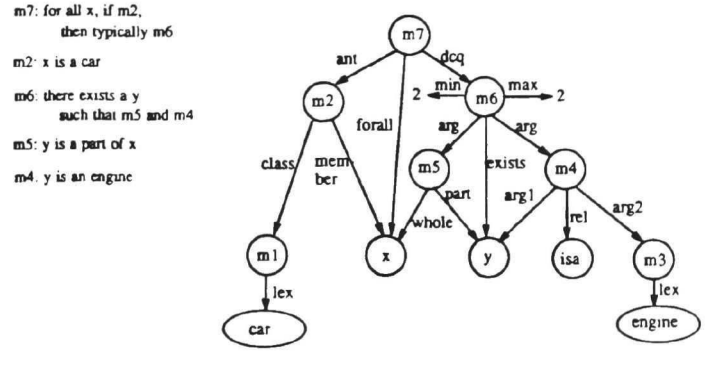
ABSTRACT

Representations for natural category systems and a retrieval-based framework are presented that provide the means for applying generic knowledge about the semantic relationships between entities in discourse and the relative salience of these entities imposed by the current context. An analysis of the use of basic and superordinate level categories in discourse is presented, and the use of our representations and processing in the task of discourse comprehension is demonstrated.

1. Introduction. We present representations for natural category systems based on a Roschian model of categories that has been extended to accommodate recent categorization research [Barsalou & Billman 1988; Keil 1989; Medin 1985, 1987; Murphy 1985, 1988, 1989]. We take issue with the assumption, implicit in most artificial intelligence (AI), natural language processing (NLP) systems, that generic concepts can be viewed as simple lists or collections of attributes. Richer representations of categories are needed to provide the intraconcept relations that structure categories and interconcept relations that provide connections to the rest of the knowledge base; these semantic relations provide some of the background or commonsense knowledge necessary for language interpretation. An analysis of the use of basic and superordinate level categories in discourse is presented, and the use of our representations and processing in the task of discourse comprehension is demonstrated. Published texts, the twenty English "Pear Story" oral narratives told by subjects after viewing a film [Chafe 1980], and forty unpublished narratives written by student subjects who were directed to retell the story of O. Henry's *A Retrieved Reformation* [unpublished data collected by Scott & Segal] provide the data analyzed in this paper.

2. Representations for Natural Category Systems. We have previously discussed the special status of the basic level in promoting inferences: the informativeness of the basic level arises from the large amount of information organized at this level and the perceptual grounding of basic level objects [Peters & Shapiro 1987ab; Peters, Shapiro, & Rapaport 1988]. In this paper we will discuss extensions to previously presented representations. Our implementation uses the SNePS knowledge representation and reasoning system, including a generalized ATN parser-generator [Shapiro 1978; Shapiro & Rapaport 1987]. Since the basic level has special status in our representations and processing, we will begin with this level.

3. Enhanced Representations for Basic Level Concepts. Default generalizations are used to represent facts about the typical members of a category in our system. Thus, a basic level category in our semantic network is, in part, a collection of default generalizations about part/whole structure, other image schematic structure, additional percepts, and functional and interactional properties [See Peters & Shapiro 1987ab, Peters, Shapiro, & Rapaport 1988 for a discussion of these structures]. Figure 1 shows a default rule that can be paraphrased as *for all x, if x is a car, then typically x has an engine*, or more simply as *typically cars have engines*. We build many such default generalizations about a basic level category such as *car*.



The following defines a path to find all the parts of basic level objects
(def-path parts (compose arg2- arg1 part- whole forall- ant class))

Figure 1

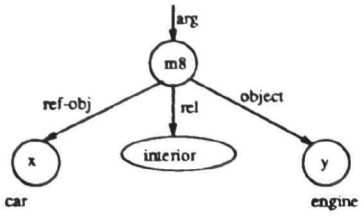


Figure 2: *Engines are interior parts of cars*

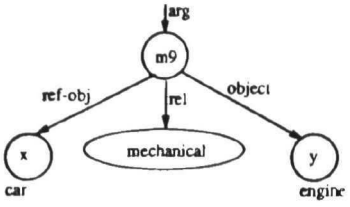


Figure 3: *Engines are mechanical parts of cars*

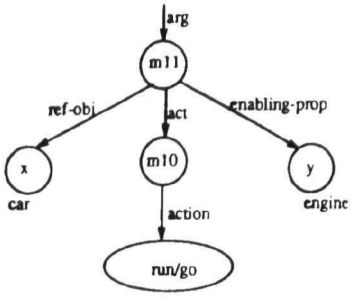


Figure 4: *Engines enable cars to run*

Many researchers have pointed out that as people's knowledge increases, they come to reject mere collections of surface attributes and other typical features as being adequate to specify concepts; categories become further structured by "deeper" conceptual relations [Barsalou & Billman 1988; Keil 1987, 1989; Medin & Wattenmaker 1987; Murphy & Medin 1985]. Additional default rules are built to capture these "deeper conceptual relations. Thus, in addition to part-whole relations (*m5* in Figure 1) and relations about other percepts, we structure basic level categories such as *car* with enabling, functional, and spatial relations, such as those shown in Figures 2-4. (We have not shown the entire default rules, just the additional conceptual relations that provide intraconcept connections. I.e, *m8*, *m9*, and *m11* would replace *m5*, the part-whole relation, in the default rule of Figure 1, creating three similar default rules.) Figure 2 shows a spatial relation that further structures (and clusters) the interior parts of *car*. We structure the external parts of *car* similarly. Figure 3 is used to further structure or cluster mechanical parts of *cars*, such as the *brakes* and *engine*; Figure 4 shows an enabling relation: *engines enable cars to run/go*. Thus, in our system, there will be many assertions linking *car* and *engine*: the knowledge associated with a basic level category such as *car* is highly interconnected and organized by spatial, temporal, casual, explanatory, and enabling relations. Basic level categories in our system are highly structured by intraconcept relations. Interconcept relations provide additional structure. E.g., *mortgage* is a thematic associate of *house*. Figure 5 shows the enabling relation that connects these two concepts.

3.1. Context-independent and Context-dependent Structure. Our representations and processing are also based on the view that the information activated after hearing or reading a category name varies widely across linguistic contexts. I.e., categorizing an entity at the basic level provides access to a large amount of information; however, only a small subset of the information associated with a category in long term memory (LTM) is incorporated in the temporary concept constructed in working memory (WM) in a particular context. Barsalou [1982] has proposed that there are two kinds of information associated with categories in LTM: context-independent (CI) properties are activated by the word for a category on all occasions, independent of context; context-dependent (CD) properties are activated only in relevant contexts.

We had originally decided that topographic structure, i.e., parts that define the overall shape of basic level objects, were context-independent attributes, i.e., automatically activated across all contexts. [Peters, Shapiro, & Rapaport 1988]. However, an examination of the normative data showing the properties listed by subjects for basic level objects in free articulation tasks [Ashcraft 1978; Rosch et al. 1976; Tversky & Hemenway 1984] disconfirms the hypothesis that all of the external parts that contribute to the overall shape are context-independent. Production frequency of properties is considered to be a measure of semantic relatedness between category names and their properties [Ashcraft 1978], and although some exterior parts of objects are always generated in these tasks, many are not. In addition, some interior (hidden) parts and some non-part attributes are always articulated. E.g., subjects generate *wheels, tires, seats, engine, steers,* and *transportation* for *car*. Here, only *wheels* and *tires* are exterior parts. *Tail, long ears, white fur, soft,* and *animal* are generated for *rabbit* (only *tail* and *long ears* are external parts), and *wings, beak, feathers, flies, eggs,* and *nests* for *bird*. We hypothesize that context-independence arises as additional causal and explanatory relations integrate or interconnect these attributes and the category name that evokes them. Thus, *engine* achieves context-independence because of its functional importance and because of the many conceptual relations that interconnect *engine* and *car*. Many interactional properties are also CI; e.g., *seats* and *steering wheel* achieve context-independence because we interact with cars by sitting on the *seats* and using the *steering wheel*. Thus, not all parts may achieve CI status, but rather, one could argue that only parts attended to get processed in a manner to produce CI status; in this case, causally relevant parts would have a distinct advantage [Barsalou 1989, personal communication].

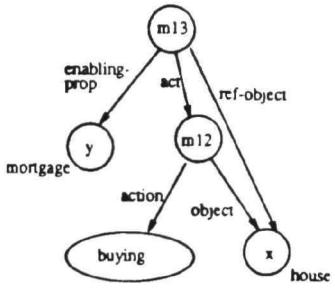


Figure 5: *Mortgages enable buying houses*

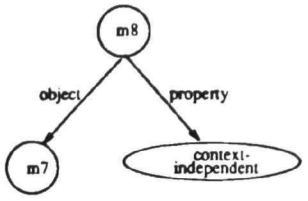


Figure 6: *Having an engine is a context-independent property of cars*

We currently create an assertion that marks CI properties, after a high degree of connectivity arises between properties and the category name. Figure 6 shows such an assertion, which we paraphrase as *having an engine is a CI property of cars*. CI properties, marked by such an assertion, are always activated when the category name is mentioned.

The category knowledge composed of the less strongly associated attributes (thematic associates and other non-centrally related entities) forms the context-dependent structure of basic level categories. We use the semantic relationships between verbs and nouns in discourse to evoke context-dependent entities, i.e., making the interrelated inferences normally made when particular verbs are used with particular nouns. E.g., *car* is associated with many different kinds of generalized actions: *driving, getting gas, washing, repairing, traveling, buying*. In our system, different knowledge associated with the category *car* is activated for each of these generalized actions. The demonstration runs in Appendix A show the activation of CI and CD entities in discourse comprehension, i.e., what interrelated inferences are made, and when they are made.

3.2. Examples of Contextual Fluctuations. Two short examples illustrate the different entities highlighted in different contexts. In the first passage, parts of the car relevant to driving, starting, and stopping are referenced:

> As the *car* crept up the slope of the bridge, the inspector burst out laughing. He laughed so hard he could scarcely give his next direction. "Stop here," he said, wiping his eyes, "then start 'er up again."
>
> Marian pulled up beside the curb. She put the car in neutral, pulled on *the emergency*, waited a moment, and then put the car into gear again. Her face was set. As she released *the brake*, her foot slipped off *the clutch pedal* and *the engine* stalled (A. Gibbs, *The Test*, p. 255, italics added).

In the second passage, in which the car is stopped, and the characters get out and move around the outside of the car, exterior parts of the car are referenced:

> Lloyd brought the *car* to a screeching halt at the very edge of the boat launching ramp.
> Lloyd looked calm, cool, and loaded as he got out and swayed his way around to *the rear* of the car. I thought he was going to open *the trunk* and present me with my father's coat, but instead he hoisted himself up on *the back* of the car. He lit a cigarette and from his perch on *the roof* he seemed to be enjoying my fright over the near plunge into the water (P. Zindel, *Confessions of a Teenage Baboon*, pp. 106-113, italics added).

The next example shows the use of activated CI information. Since *fur* is automatically activated for Cyril, the cat, (fur has CI status, since it is an interactional property; i.e., people pet cats, stroking their fur), disambiguation of the reference *his fur* is easily handled. Disambiguation here does not involve choosing the most activated or highly focused entity, but rather choosing an activated entity with this activated property. Claws also has CI status for cats.

> When Jury {a detective} opened the door to Racer's {Jury's boss} sanctuary, Cyril {cat} slid between *his feet*, streaked snake-like across a carpet the color of *his fur*, and was scaling the bookcase set back against the wall to the left of Racer's large desk. *His claws* were like pinions digging into forensic science, ... (M. Grimes, *The Five Bells and Bladebone*, p. 188, italics added).

An example showing a reference to a CD entity follows. Here *the paw* is evoked in the context of the cat *striking* at the water:

> The cat walked out of the secluded garden and toward the bank of a stream farther on. Here it crouched and watched a wren having a dust bath. Before it could pounce, the wren was away, skimming across the water. Looking into the stream, as if the bird might have fallen there, the cat saw shadows deep inside darting, hanging suspended, darting forward again. The cat struck at the water, trying to fix the moving shadow. ... It yawned again, washed at *the paw*, stopped when it saw something skittering across the footbridge and followed. (M. Grimes, p. 11, italics added)

4. Superordinates and Basic Level Names in Discourse. Since basic level categories carry the most information, they denote referents at their level of "usual" utility; i.e., at a level that is both sufficiently informative and cognitively efficient to manipulate. Thus the use of basic level terms to refer to entities in discourse follows the Gricean

maxims of conversational quantity and manner, since basic level names usually carry sufficient information for an addressee/reader to be able to identify the individual or category being referred to. What constraints cause a speaker/writer to abandon basic level terms in categorizing a given entity at a given point? In particular, when are superordinate level names used in discourse, and how is the knowledge stored with superordinate level concepts used during discourse comprehension? The next sections will analyze the use of superordinate terms in written and oral narratives.

4.1. Discourse Analysis: Use of Superordinate Level Names. Basic level terms constituted 93% of the nominal references to concrete entities in the twenty "Pear Story" oral narratives; superordinate level terms were used very infrequently, constituting only 2% of the references [Downing, 1980]. Basic level terms again predominated in the Scott & Segal subjects' written narratives, constituting 75% of nominal references to concrete entities. Superordinates occurred much more frequently than in the oral "Pear Story" narratives, constituting 17% of the references in the Scott & Segal written narratives. In the next sections superordinate level references will be examined more carefully.

4.1.1. Groups, Collections, and Classes. It has been suggested [e.g., Murphy & Wisniewski 1989; Wisniewski & Murphy 1989] that superordinates are frequently used to refer to groups, collections, and classes. I.e., basic level names are frequently used to refer to individuals or single objects, while speakers use superordinate names such as *furniture, clothes* and *plants* to refer to groups of related objects simultaneously [Wisniewski & Murphy 1989]. Our text analysis confirmed this usage of superordinate names. The following passages from O. Henry's *A Retrieved Reformation* (with italics added to indicate the superordinate level term) illustrate this usage:

(1) Take him back, Cronin, smiled the warden, and fix him up with outgoing *clothes*.

(2) Pulling out from the wall a folding-bed, Jimmy slid back a panel in the wall and dragged out a dust-covered suitcase. He opened this and gazed fondly at the finest set of burglar's *tools* in the East.

(3) He was at much at home in the *family* of Mr. Adams and that of Annabel's married sister as if he were already a member.

In the following sections we discuss additional uses of superordinate labels that were found in the written and oral narratives examined.

4.1.2. Introductory Mentions of Entities. Frequently superordinate labels are used to introduce entities to the hearer/reader, i.e., an author/speaker uses a superordinate level term for the first mention of an entity, later switching to a basic level term [Downing, 1980]. This technique is typically used only with characters and other elements that are central to the narrative. Two examples of this technique follow:

It was on the Dover road that lay, on a Friday night late in November, before the first of the *persons* with whom this history has business. [Dickens, *A Tale of Two Cities*, p. 8]

About three hundred eighty-five thousand years ago, when the oceans and continents were in place as we know them today, the land bridge from Asia was open, and a huge ponderous *animal*, looking much like an oversized elephant but with enormous protruding tusks, slowly made his way eastward, followed by four females and their young. [Michener, *Alaska*, p. 15]

In both of these examples the reader has the expectation that she will soon find out more about the characters being introduced by the superordinate level terms, and the superordinate level names seem to contribute to the readers being gradually "drawn" into the story. Thus, the readers' attention becomes focused on entities introduced in this way.

4.1.3. References to Focused Discourse Entities. A superordinate label is also frequently used to refer to a discourse entity that is focused, but not so highly activated that a pronoun or zero is easy to understand, yet too highly activated for a basic level term to be used without sounding childish or redundant. I.e., the basic level term may give too much information, given the focus level of the referent. For example:

He took a train for three hours, got off and went to a cafe owned by Mike Dolan. Dolan gave Jimmy his key. Jimmy went to a room in the back of the *establishment*. [Scott & Segal unpublished data, subject's retelling of O. Henry's, *A Retrieved Reformation*]

At the end of this passage the key and the cafe are competing, focused discourse entities (both can be referred to using *it*). Since the key is more highly focused, a pronominal reference to the cafe (use of *it*) seems less suitable than the use of the superordinate term *the establishment*. The use of the basic level term *the cafe* also seems less suitable than *the establishment*, because it is redundant. An additional example follows:

The cab stopped in front of Lloyd's house and I got out. I think the driver thought I was going to run away without paying but my slacks were so tight I couldn't get my money out of my pocket without standing up. I gave him the fare, including a good tip, which left me with about a dollar and twenty-nine cents to my name. It was the absolute end of

my savings from the last case my mother had, where her boss gave me a couple of dollars a day to walk his Yorkies. The driver just floored his junk heap and took off making so much noise I didn't hear the lilt of music until the taxi turned at the far corner. It had been quite a while since Helen and I had vacated the *place* and I couldn't believe the party was still going on. [P. Zindel, *Confessions of a Teenaged Baboon, p. 141]*

Here, a pronominal reference to the house (*it*) cannot be used, because there is too much distance between the first mention of the *house* and this reference. However, the superordinate, *the place*, is easily disambiguated. The next example again shows competing, focused discourse entities:

Anyway, this time it was a black cat reclining in the middle of this busy boulevard and it was positioned so it looked like it had been half run over by a car, squashing part of its body into a type of base like a buttress for a paper doll. It looked like I was staring at a stone statue of a cat with metallic eyes glazed with fire and that if I wanted to I could just take the *thing* home and use it as a garden ornament. The *animal* seemed dead except for its eyes, and I had the strangest feeling it might still be alive, . . . [P. Zindel, *Confessions of a Teenaged Baboon, p. 121]*

Here the superordinate label *the thing* refers to one discourse entity, the *stone statue of a cat*, while *the animal* refers to a second discourse entity, the *cat.*

4.1.4. Superordinates: Generality. Since superordinate labels are more general (less specific) than basic level terms, they can be used to communicate a sense of "vagueness". Thus, a superordinate name may be used to communicate that someone (a) doesn't know what something is, or what the name of something is (see example (1) below); (b) cannot see something very well (see example (2) below; (c) doesn't remember specifically what something was (see example (3) below); (d) doesn't want to call attention to irrelevant details or promote too many inferences (see example (3) again), or (e) wants to deliberately conceal information (e.g., in a mystery the detective may choose not to reveal the specific murder weapon used to commit the crime to a witness being questioned).

(1) Three boys came out, helped him pick himself up, pick up his bike, pick up the pears, one of them had a *toy*, which was like a clapper. And, I don't know what you call it except a paddle with a ball suspended on a string. [Chafe, 1980, transcript from English "Pear Film" narrative]

(2) Zoe squinted through a square of the lattice. Outside the crawl space it was lighter. She could see three *figures* standing beside a small truck. [Z. Oneal, *War Work, p. 133]*

(3) He [Jimmy] jumps on a train heading for the state boarder [sic] after he has some *food* and wine not of prison origin. [Scott & Segal unpublished data, subject's retelling of O. Henry's, *A Retrieved Reformation*]

In (1) the speaker telling the story of the "Pear Film" didn't know a name for the toy that a character in the film was playing with. In (2) Zoe cannot see the characters she is describing very well. In (3) the subject retelling the story either didn't remember that the food Jimmy Valentine had eaten was broiled chicken, or chose to be less specific when recounting this event, using a superordinate label rather than a more specific, basic level one.

4.2. Non-taxonomic Superordinates. In general, superordinates convey much less information than basic level terms. However, non-taxonomic superordinates (e.g., friend, enemy, jerk) frequently add information, communicating an attitude toward the referent(s):

I'm fifty and for half of those years I had worked for a Chicago newspaper I would rather not identify because it no longer exists. Its name exists, but it is a vulgar *rag* filled with sex and crime, edited by a gang of *creeps*, and owned by a right wing egomaniac who bought it for a song from the childish, idiotic woman who inherited it. [D. Kiker, *Murder on Clam Pond*, p. 22]

Non-taxonomic superordinates also frequently focus attention on particular attributes. In the following example, *enemy* evokes or activates entities such as *claws, tusks, teeth* as well as external body parts. All of these are relevant in the context of the struggle between these two enemies.

. . . he [Mastodon] heard a rustle that disturbed him. Prudently, he withdrew lest some enemy leap upon him from a hiding place high in the trees, and he was not a moment too soon, for as he turned away from the willow, he saw emerging from the protection of a nearby copse *his most fearsome enemy.*

It was a kind of tiger with powerful *claws* and a pair of frightful upper *teeth* almost three feet long and incredibly sharp. Mastodon knew that though this saber-tooth could not drive *those fearsome teeth* through *the heavy skin of his protected rear or sides*, it could if it obtained a secure foothold on *his back*, sink them into *the softer skin at the base of his neck.* . . . Mastodon had *his long tusks*, of course, but he could not lunge forward and expect to impale his adversary on them, they were not intended for this purpose. [J. Michener, *Alaska*, p. 17, italics added]

4.3. Superordinates: Theories and Causal Reasoning. A number of researchers [e.g., Keil 1987, 1989; Murphy & Medin 1985; Barsalou & Billman 1988] have pointed out that our deep, rich theories about objects are linked to superordinate level concepts: theories that take us beyond categorization based on perceptual or surface similarity. In

order to appropriately categorize a whale as a mammal, we are required to move beyond using surface similarity as a basis for categorization: we need to recognize that hidden features and elaborated theories about origins are more important than perceptual features in categorizing biological kinds at a superordinate level. Theoretical knowledge also helps us to deduce reasons for correlations among basic level features: e.g., wings, hollow bones, small, and flies are correlated because they are causally linked. Our scientific knowledge organized at the superordinate level helps us understand these linkages. Taxonomic superordinates enrich basic level concepts, providing access to deep causal and explanatory relations.

4.4. Causal Reasoning in Discourse Understanding. Even when superordinates are not explicitly mentioned in discourse, they may provide the causal and explanatory relations necessary to our understanding of discourse. Our understanding of humor frequently depends on our use of superordinate level knowledge. E.g., in a short monologue about coffee, Garrison Keillor stated that although people said that decaffeinated coffee was dangerous and that a chemical used to decaffeinate coffee caused cancer in rats, his feeling was that the more rats that get cancer the better [*Garrison Keillor's American Radio Company of the Air*, February 16, 1990, WNED-FM]. This joke like many others is amusing because we recognize that an inappropriate conclusion has been drawn: the reasoning has been ineffectual. I.e., we know that rats, like people, are animals (in fact we know that both are mammals); that animal models are used to screen carcinogenic agents (with rats frequently being used as test animals); that if a chemical causes cancer in one kind of animal, there is a great likelihood that it will cause cancer in other animals (especially closely related animals). The knowledge used in this reasoning is superordinate level knowledge.

5. Discourse Processing: Inferences. The processing performed by our system will be illustrated by considering the problem of comprehending references to implicitly evoked CI and CD entities in discourse. As stated previously, basic level categories evoke a rich set of entities which may be referred to later in discourse. Superordinates, in contrast, evoke few discourse entities directly associated with the superordinate category itself, but frequently influence what is activated

The inferences made at the basic level seem to involve automatic processing: i.e., they are initiated by well-established memory structures, rely on connectivity between concepts, are computationally "cheap", and are plausible inferences rather than logically derived assertions which necessarily follow from the text. Superordinates, in contrast, do not seem to promote many inferences [Rosch 1981; Tversky & Hemenway 1984; Gelman 1989]: subjects frequently list few or no attributes for superordinate level concepts. Thus, our knowledge about superordinate concepts is a deeper and less easily verbalized knowledge, involving underlying principles and theories about the world. Inferencing at this level is not computationally "cheap" and effortless, but instead seems to require substantial attentional resources. The inferences made involve causal knowledge and explanatory relations (theoretically-oriented knowledge), and are deducible from the text and the knowledge base. We use a "spreading activation" form of inference similar to marker passing to activate the relevant basic level information, and deductive inference to reason about theory-laden superordinate level knowledge, forming assertions which follow logically or necessarily from the text.

5.1. Path-Based Inference. The SNePS path-based inference package provides the subconscious reasoning required for the automatic activation of CI and relevant CD entities: the definition of appropriate paths in the network enables the automatic retrieval of the relevant satellite concepts of basic level concepts. The context-independent entities can be retrieved by defining a path of arcs from a node representing a basic level category, e.g., *car*, to its context-independent properties: *wheels, tires, engine, seats, steering wheel.* The following path is used to retrieve all context-independent properties of basic level categories:

(find (compose arg2- arg- deq- object- (domain-restrict (property context-independent)) object ant class)(find lex *basic-level-category-name*))

Our current implementation activates context-dependent information in response to discourse comprehension of events containing generalized actions such as *buying/selling a house, driving/stopping/repairing/washing a car, seeing/walking/washing a dog.* Additional paths are defined to retrieve these entities. Parts can be retrieved by defining a path of arcs from a node representing a basic level category, e.g., *car*, to its parts: *engine, roof, hood, trunk, wheels, brakes*, etc. Figure 1 in section 3 shows a defined path called *parts* which is used to activate these entities. The "parts" path is never used alone, rather it's used in conjunction with additional paths specified through default rules such as those shown in part in Figures 2-4, i.e., paths through the spatial, enabling, and other intraconcept and interconcept relations described earlier. E.g., we use an "exterior or surface parts path" to retrieve only exterior parts following generalized actions such as *washing a car* or *watching a bird*, and interior, mechanical parts following generalized actions such as *repairing a car*. Causal and enabling relations are considered to be more important than other relations in this system, and satellite associates that enable actions are activated by using "paths to enablers" to retrieve these associates in the relevant contexts. E.g., *mortgage* is retrieved following the generalized action, *buying a house*. An ATN grammar then makes use of the defined paths to activate, i.e., implicitly focus, the context-

independent and relevant context-dependent satellite entities of a basic level category (that has been either encountered in input or activated by a subordinate level concept): returning all the nodes that are found at the end of the defined paths of arcs emanating from the basic level categories and placing them in the system's working memory. (It is well established that categories automatically activate their superiors [Rosch 1978; Barsalou 1982].)

6. Demonstration of Discourse Processing. Appendix A shows four sample runs of SNePS/CASSIE illustrating some of our current capabilities. User input is on lines with the **:-prompt**; the systems' output and timing information are on the following lines. After a few of the sentences, a list of the activated context-independent (CI) and context-dependent (CD) entities or nodes (labeled CI evoked and CD evoked respectively) associated with the basic level category *car* is shown. (A complete listing of the associates of *car* activated in these sample runs is also found in Appendix A. See *Nodes Evoked in Demos 1, 2, & 3.*)

In sentence (1abc), comprehension of the basic level category *car* implicitly evokes many entities, including the following CI entities: the *engine, seats, steering wheel, wheels,* and *tires.*[1] In addition, the verb *bought* in conjunction with *car* activates a CD entity: the *price* of the *car.* Sentence (2a) of Demo 1 contains a reference to an implicitly focused CI item, *the engine*; sentence (3a), a reference to an implicitly focused CD item, *the price.* The verb *fixed* in conjunction with *car* in sentence (4a) of Demo 1 causes the activation of many CD associates of *car*, e.g., the *carburetor, distributor, spark plugs, battery,* and *brakes.*[2] Sentences (5a) - (7a) contain references to previously activated CI and CD entities.

In sentence (2b) of Demo 2, the generalized action *washing the car* causes activation of many CD entities: exterior parts of cars (the *bumper, grill, trunk, hood, roof, tires, wheels, windshield, finish,* and the *exterior*). Many of these activated entities are referenced in sentences (3b) - (7b).

In sentence (4c) of Demo 3, the mention of *starting the car* evokes many CD entities, e.g., the *ignition, key, battery, accelerator, brakes,* and a CI entity, the *engine.* Sentences (5c) - (12c) contain references to these activated entities.

In sentence (1d) of Demo 4 comprehension of the basic level category *house* implicitly activates many CI entities (e.g., *windows, doors, roof, shelter*). In addition, the verb *bought* in conjunction with *house*, activates such CD entities as the *mortgage* and *price.* Thus, the concept *house* constructed in WM has been tailored to the current context. Sentence (2d) contains a reference to the *mortgage* which was activated in (1d). Sentence (4d) contains a reference to *the butler*, an entity that has not been activated, since it is not a CI associate of *house*, and was not activated in the context of *buying a house.* Thus, comprehension of *the butler* requires inferencing using the total knowledge base. A comparison of the timing information for sentences (2d) and (4d) illustrates that in our system the comprehension time for a non-activated entity (*the butler*) is longer than that for an activated entity (*the mortgage*). Sentence (5d) contains a reference to the subordinate level category *collie*, which activates its immediate superior, the basic level category *dog.* The usual inferences about *dogs* are then drawn, and many CI entities (e.g., *tail, barking, animal*) are activated. It also contains the generalized action of *buying a dog*, so *cost* is activated. In sentence (6d), resolution of the definite anaphor *her barking* cannot be based on the normal mechanism of finding the most highly focused antecedent that matches the semantic features of the possessive pronoun *her.* It was not Lucy's barking! Rather, it requires a search of WM for the concept of *barking*, returning its evoking concept *collie.* Sentences (12d) and (13d) also illustrate the need for using a focusing mechanism based on more than activatedness, recency, and matching semantic features. I.e., *tail* was evoked as part of the basic level category *cat* in (7d), but not as part of either the subordinate level category *canary* or its immediate superior *bird*; whereas *chirp* was evoked as associated with *bird/canary* in (9d), but not with *cat.* Our processing of *his tail* and *his chirp* simply involves searching WM for these previously activated entities and their evoking concepts, not a search of the whole knowledge base. Thus, integration of implicitly evoked associates (e.g., *chirp*) with the previously mentioned evoking concept (e.g., *bird/canary/Tweety*) is quite simple and comprehension time quite fast.

---

[1] The CI entities that are always automatically activated in our system are those associates that have many causal and explanatory relations integrating them with the basic level category name that evokes them. They also are taken from property-norm data of Ashcraft, Rosch, and Tversky & Hemenway referenced earlier in this paper.

[2]Because of space limitations for these sample runs, only a few CD entities are shown as being evoked. Many more nodes are actually activated.

7. Future Research. Many problems remain to be solved. In particular, further work is needed in developing representations for and using superordinate level knowledge, in deciding which properties associated with a category have context-independent status, and in extending our work to consider context effects arising from the topic area, the task at hand, goals and discourse purposes.

REFERENCES

(1) Ashcraft, M. (1978), "Property Norms for Typical and Atypical Items from 17 Categories: A Description and Discussion," *Memory & Cognition*, vol 6, no. 3, pp. 227-232.

(2) Barsalou, L. W. (1982), "Context-independent and context-dependent information in concepts," *Memory & Cognition*, vol 10, pp. 82-93.

(3) Barsalou, L. W. (1987), "The Instability of Graded Structure," In U. Neisser (ed.), *Concepts and Conceptual Development* (Cambridge: Cambridge University Press).

(4) Barsalou, L. W., & Billman, D. (1988), "Systematicity and Semantic Ambiguity," In D. S. Gorfein (ed.), *Resolving Semantic Ambiguity* (New York: Springer-Verlag).

(5) Downing, P. (1980), "Factors Influencing Lexical Choice in Narrative," in W. Chafe, (ed.), *The Pear Stories* (New Jersey: ABLEX Publishing Company)

(6) Keil, F. (1987), "Conceptual Development and Category Structure," In U. Neisser, (ed.), *Concepts and Conceptual Development* (Cambridge: Cambridge University Press).

(7) Keil, F. (1989), *Concepts, Kinds, and Cognitive Development* (Cambridge: MIT Press)

(8) Mervis, C. B., & Rosch, E. (1981), "Categorization of Natural Objects," Ann. Rev. Psychol., vol. 32, pp. 89-115.

(9) Medin, D., & Wattenmaker, W. (1987), "Category Cohesiveness, Theories, and Cognitive Archeology," In U. Neisser, (ed.), *Concepts and Conceptual Development* (Cambridge: Cambridge University Press).

(10) Murphy, G., & Medin, D. (1985), "The Role of Theories in Conceptual Coherence," *Psychological Review*, vol. 92, pp. 289-316.

(11) Murphy, G. L., & Wisniewski, E. J. (1989), "Categorizing Objects in Isolation and in Scenes: What a Superordinate is Good For," *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 15, 572-586.

(12) Peters, S. L., & Shapiro, S. C. (1987a), "A Representation for Natural Category Systems I," *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, Seattle, WA, pp. 379-390,

(13) Peters, S. L., & Shapiro, S. C. (1987b), "A Representation for Natural Category Systems II," *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, Milan, pp. 140-146.

(14) Peters, S. L., & Shapiro, S. C. (1988), "Flexible Natural Language Processing and Roschian Category Theory," *Proceedings of the Tenth Annual Conference of the Cognitive Science Society*, Montreal, Quebec, Canada, pp.

(15) Rosch, E., Mervis, C. B., Gray, W. D., Johnson, D. M., & Boyes-Braem, P. (1976), "Basic Objects in Natural Categories," *Cognitive Psychology*, vol. 8, pp. 382-439.

(16) Rosch, E., & Lloyd, B. B., (eds.), (1978), *Cognition and Categorization* (Hillsdale, NJ: Lawrence Erlbaum Associates).

(17) Shapiro, S. C. (1978), "Path-Based and Node-Based Inference in Semantic Networks," In D. Waltz, (ed.), *Theoretical Issues in Natural Language Processing-2* (Urbana, Illinois) pp. 219-225.

(18) Shapiro, S. C., & Rapaport, W. J. (1987), "SNePS Considered as a Fully Intensional Propositional Semantic Network," In G. McCalla & N. Cercone, (eds.), *The Knowledge Frontier: Essays in the Representation of Knowledge* (New York: Springer-Verlag), pp. 262-315.

(19) Tversky, B., & Hemenway, K. (1984), "Objects, Parts, and Categories," *Journal Of Experimental Psychology: General*, vol. 113, pp. 169-93.

(20) Wisniewski, E. J., & Murphy, G. L. (1989) "Superordinate and Basic Category Names in Discourse: A Textual Analysis," *Discourse Processes*, 12, 245-261.

CITED TEXTS

(1) Dickens, C., *A Tale of Two Cities* (New York: Colonial Press)

(2) Gibbs, A., (1940), "The Test," in *Short Stories from the New Yorker* (New York: Simon and Schuster), 252-256.

(3) Grimes, M., (1987), *The Five Bells and Bladebone* (New York: Dell).

(4) Kiker, D. (1986), *Murder on Clam Pond* (New York: Ballantine Books).

(5) Michener, J., (1988), *Alaska* (New York: Random House).

(6) North, S., (1963), *Rascal* (New York: E.P. Dutton & Co.).

(7) Scott, P., & Segal, E., Unpublished Narratives, Retellings of O. Henry's, "A Retrieved Reformation".

(8) Zindel, P., (1977), *Confessions of a Teenage Baboon* (New York: Harper and Row).

Appendix A

Demo 1: *Repairing the Car*

(1a) : Lucy bought an old car.
I understand that Lucy bought an old car
CI evoked: (m138 m51 m20 m12 m3 m317)
CD evoked: (m94)
( ) exec: 12.41 sec gc: 4.71 sec

(2a) : the engine was rebuilt.
I understand that the engine of the old car is rebuilt
( ) exec: 9.28 sec gc: 2.35 sec

(3a) : the price was low.
I understand that the price of the old car is low
( ) exec: 9.21 sec gc: 2.33 sec

(4a) : A mechanic fixed the car.
I understand that a mechanic fixed the old car
CD evoked:(m245 m224 m217 m153 m35 m3)
( ) exec: 13.08 sec gc: 2.43 sec

(5a) : He tuned the engine.
I understand that the mechanic tuned the engine of the old car
( ) exec: 11.86 sec gc: 4.85 sec

(6a) : He replaced the carburetor.
I understand that the mechanic replaced the carburetor of the old car
( ) exec: 11.81 sec gc: 2.38 sec

(7a) : He fixed the distributor.
I understand that the mechanic fixed the distributor of the old car
( ) exec: 11.86 sec gc: 2.40 sec

Demo 2: *Washing the Car*

(1b) :Lucy bought a red car
I understand that Lucy bought a red car
CI evoked: (m138 m51 m20 m12 m3 m317)
CD evoked: (m94)
( ) exec: 12.38 sec gc: 4.83 sec

(2b) : She washed the car.
I understand that Lucy washed the red car
CD evoked: (m278 m265 m131 m87 m80 m73 m66 m59 m28 m20 m12)
( ) exec: 11.88 sec gc: 2.35 sec

(3b) : She cleaned the windshield.
I understand that Lucy cleaned the windshield of the red car
( ) exec: 11.98 sec gc: 4.83 sec

(4b) : She washed the grill.
I understand that Lucy washed the grill of the red car
( ) exec: 11.75 sec gc: 4.81 sec

(5b) : She scrubbed the front bumper..
I understand that Lucy scrubbed the front bumper of the red car
( ) exec: 12.36 sec gc: 4.98 sec

(6b) : She waxed the hood.
I understand that Lucy waxed the hood of the red car
( ) exec: 12.21 sec gc: 2.63 sec

(7b) : The exterior is sparkling.
I understand that the exterior of the red car is sparkling
( ) exec: 10.55 sec gc: 2.48 sec

Demo 3: *Driving the Car*

(1c) : Lucy bought an old car.
I understand that Lucy bought an old car
CI evoked: (m138 m51 m20 m12 m3 m317)
CD evoked: (m94)
( ) exec: 12.73 sec gc: 4.80 sec

(2c) : she took a driving test.
I understand that Lucy took a driving test
( ) exec: 10.36 sec gc: 4.73 sec

(3c) : She was nervous.
I understand that Lucy is nervous
( ) exec: 6.18 sec gc: 2.40 sec

(4c) : she started the car.
I understand that nervous Lucy started the old car
CD evoked: (m224 m160 m147 m43 m35 m3)
( ) exec: 12.51 sec gc: 2.41 sec

(5c) : she flooded the engine.
I understand that nervous Lucy flooded the engine of the old car ( )
exec: 13.01 sec gc: 4.85 sec

(6c) : she restarted the car.
I understand that nervous Lucy restarted the old car
( ) exec: 11.33 sec gc: 2.41 sec

(7c) : she tapped the accelerator.
I understand that nervous Lucy tapped the accelerator of the old car ( )
exec: 12.90 sec gc: 4.95 sec

(8c) : she entered a busy street.
I understand that nervous Lucy entered a busy street
( ) exec: 12.30 sec gc: 4.86 sec

(9c) : she approached an intersection.
I understand that nervous Lucy approached an intersection
( ) exec: 10.25 sec gc: 2.48 sec

(10c) :she stopped the car.
I understand that nervous Lucy stopped the old car
( ) exec: 13.15 sec gc: 2.58 sec

(11c) : the brakes were squeaky.
I understand that the brakes of the old car are squeaky
( ) exec: 8.25 sec gc: 0.00 sec

(12c) : she failed the test.
I understand that nervous Lucy failed the driving test
( ) exec: 11.03 sec gc: 2.58 sec

Demo 4: General Demo

(1d) : Lucy bought a Victorian house.
I understand that Lucy bought a Victorian house
( ) exec: 11.28 sec gc: 4.83 sec

(2d) : The mortgage is huge.
I understand that the mortgage of the Victorian house is huge
( ) exec: 9.28 sec gc: 2.35 sec

(3d) : John visited her.
I understand that John visited Lucy
( ) exec: 8.48 sec gc: 2.48 sec

(4d) : The butler opened the door.
I understand that a butler opened the door of the Victorian house ( )
exec: 20.83 sec gc: 7.45 sec

(5d) : Lucy bought a collie.
I understand that Lucy bought a collie
( ) exec: 11.83 sec gc: 2.61 sec

(6d) : Her bark wakes Lucy.
I understand that the bark of the collie is waking Lucy
( ) exec: 13.98 sec gc: 5.16 sec

(7d) : John owns a cat
I understand that John owns a cat
( ) exec: 10.20 sec gc: 2.46 sec

Demo 4 Continued

(8d) : The cat is named Sylvester.
I understand that Sylvester is the cat
( ) exec: 8.05 sec gc: 0.00 sec

(9d) : He bought a canary.
I understand that John bought a canary
( ) exec: 10.66 sec gc: 2.48 sec

(10d) : The bird is named Tweety.
I understand that Tweety is the bird
( ) exec: 8.60 sec gc: 0.00 sec

(11d) : The cat stalks Tweety.
I understand that Sylvester is stalking Tweety
( ) exec: 11.86 sec gc: 2.36 sec

(12d) : His tail is swishing.
I understand that the tail of Sylvester is swishing
( ) exec: 10.63 sec gc: 2.53 sec

(13d) : The chirp alerted John.
I understand that the chirp of Tweety alerted John
( ) exec: 13.46 sec gc: 2.58 sec

Nodes evoked in Demos 1, 2, & 3:

(m3 (lex (engine)))
(m12 (lex (tire)))
(m20 (lex (wheel))) (m28 (lex (windshield)))
(m35 (lex (brake))) (m43 (lex (accelerator)))
(m51 (lex (seat))) (m59 (lex (trunk)
(m66 (lex (roof))) (m73 (lex (hood))
(m80 (lex (exterior)))
(m87 (lex (finish)) (m94 (lex (price)))
(m131 (lex (door)))
(m138 (lex (steeringwheel)))
(m147 (lex (ignition)))
(m153 (lex (carburetor)))
(m160 (lex (key)))
(m217 (lex (sparkplugs)))
(m224 (lex (battery)))
(m245 (lex (distributor)))
(m265 (lex (grill)))
(m278 (lex (bumper)))
(m317 (lex (vehicle)))

Learning Overlapping Categories*

Joel D. Martin
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, Georgia 30332
joel@gatech.edu

Abstract

Models of human category learning have predominately assumed that both the structure in the world and the analogous structure of the internal cognitive representations are best modeled by hierarchies of disjoint categories. Strict taxonomies do, in fact, capture important structure of the world. However, there are realistic situations in which systems of overlapping categories can engender more accurate inferences than can taxonomies. Two preliminary models for learning overlapping categories are presented and their benefit is illustrated. The models are discussed with respect to their potential implications for theory-based category learning and conceptual combination.

1 Introduction

The natural world can be neatly organized into a hierarchy of disjoint categories. A platypus is a mammal, not a bird; and mammals are animals, not plants. Artificial categories can also be placed in a strict hierarchy. Something that is a balpeen hammer is not a claw hammer. Similarly, it is a hammer, not a saw; and a hand tool, not a power tool. Each category, whether natural or artificial, can be distinguished from its alternatives by its distinct intensional description, which might be represented as lists of feature frequencies or a list of instances. Taxonomies such as these are elegant, economical, and seem to be the most suitable representation for many natural and artificial categories.

Presumably, if humans learn and use taxonomic structures, they would efficiently and accurately characterize the regularities in the world. The assumption that they do so has been an extremely fruitful seed for generating models of human category learning. So fruitful, in fact, that alternate structures of categories have been neglected until recently. Structures of overlapping categories can actually be superior to taxonomies when there are multiple equally good ways to partition a set of experiences. In these circumstances, non-disjoint categories permit faster learning and more economical storage than is possible with disjoint categories.

If one assumes the world contains such domains and that humans are optimally adapted to their environment as suggested by Anderson (1988), then overlapping category structures should motivate better models of human behavior. Indeed, some structures of overlapping categories suggest preliminary methods for modeling theory-based concept learning (Murphy & Medin, 1985) and conceptual combination (Smith & Osherson, 1984).

*This research was supported in part by a Georgia Institute of Technology Stelson grant and NIH grant 7R23HD20522-03.

2 Taxonomies

The purpose of taxonomies is to maximize the probability of correct inference without requiring the retrieval of large sets of past experiences for every prediction. They compactly summarize predictive relationships. However, there are domains for which taxonomies not only do not maximize the probability of correct inferences, but also do not compactly represent the regularities. These are domains that permit multiple different, but equally predictive taxonomies. Each of these taxonomies allows roughly the same number of correct inferences, but they differ with respect to which attributes or aspects of experiences about which they are most informative.

2.1 Disjoint categories

A set of objects or events can be split into disjoint subsets in many ways. Given a collection of a dog, a cat, a goldfish, and a whale, one split may group the dog with the fish and the cat with the whale. Another may assign each individual to a separate subset, and yet another may divide the mammals from the fish. Of these many possibilities the one that permits the most correct inferences is preferred. Any collection of categories is only useful to the extent that category membership permits accurate predictions. For instance, knowing that a particular object is a bird allows probable predictions about body covering, food, and habitat.

This notion has been formalized for a single set of disjunctive categories by both Gluck and Corter (1985) and Anderson (1990). For example, Gluck and Corter derive that the probability of making correct inferences is equal to the sum, over all categories, of the probability of the category multiplied by the square of the probability of a particular value given the category:

$$P(correct) = \sum_k P(C_k) \sum_j \sum_i P(A_j = V_{ij} \mid C_k)^2 \qquad (1)$$

This measure could be applied by performing every possible partitioning and choosing the one with the largest $P(correct)$. This would be horrendously expensive, though, because the number of possible partitionings grows exponentially with the number of categories. A more practical approach accepts a set of experiences in a particular order and deals with each instance in turn. Each instance can be used to update the intension of a category. It is added to the category that allows the greatest rise in the probability of correct inference. This is a hill-climbing search toward the optimal partitioning (Fisher, 1987).

Gluck and Corter (1985) originally intended their measure to be a predictor of the basic level in a taxonomy (eg. Rosch et al., 1976). However, it or a similar measure can be used recursively to form complete taxonomies, as was suggested by Fisher (1987) and Anderson (1990).

Gluck and Corter's measure will be used below as a paradigmatic example of a model for learning disjoint categories. As well, Fisher's (1987) method for generating hierarchies will be assumed. The generated hierarchies, unlike the disjoint category measure, are not necessarily close to optimal. The technique produces performance that is below the theoretical maximum in some situations because it attempts to maximize the predictive work done at each layer. However, there are no formal theories for acquiring optimally predictive hierarchies, so this heuristic method will suffice.

2.2 The trouble with taxonomies

Fisher's (1987) COBWEB system, which is rooted in Gluck and Corter's (1985) measure, has been quite successful at acquiring taxonomies that allow accurate inferences. COBWEB has been applied to both real and artificial domains. Given this success, it is valuable to identify what

shortcomings COBWEB and related models might have, if any. As well, if shortcomings are found, it is essential to demonstrate that the conditions in which taxonomies fail occur frequently in human experience. However, if such conditions are very rare, humans would do well enough with taxonomies and not require any alternative model.

2.2.1 The problems

All of the difficulties with taxonomies that will be discussed here occur when the domain to be learned allows many different, but equally useful taxonomies. These domains will be referred to as *cross-classification* domains. For example, a list of athletes may specify their height, weight, eye color, and hair color. For these athletes, the values for height and weight are mutually predictive, as are the values for eye and hair color. Height though, is only randomly related to eye and hair color, and similarly for weight. Taller athletes are heavier and blue-eyed ones tend to be blond, but tall athletes do not have a characteristic hair color.

A system that learns disjoint categories in this domain has several choices. First, it could choose to partition the instances on the basis of height and weight or on the basis of eye and hair color. This, however, limits the probability of correct inferences by completely ignoring one of the predictive relationships. Alternatively, it could try to balance the two relationships without storing every instance. This again loses some predictive information. In order for the categories to capture one relationship, it must jeopardize the accuracy of the other.

A third possibility is that the system could simply create a category for each combination of values for the domain. This results in no timing or storage benefit for categories over individual instances, but does achieve better prediction than the above proposals. This method will be generally inefficient for domains in which there are multiple clusters of mutually predictive values, because there is potentially an exponential number of categories that must be stored. More importantly, though, each of these categories must be separately learned. This requires that every combination must be seen before the domain is well learned. This is likely too restrictive, because humans, at least, do not have to have seen a striped apple to be able to make inferences about it.

A final possibility is that the system could choose one relationship for each level of the hierarchy, such that, for example, height and weight would initially classify the instance, and then hair and eye color would classify it further. This possibility shares the criticisms above, but is somewhat more economical. Instead of storing every combination of values as a separate category, this possibility allows each partitioning to be a decision about the values of some set of correlated attributes, thereby apparently reducing the order of the number of categories. However, each decision except the one at the top of the tree must be replicated many times throughout the hierarchy. This requires excessive storage and, as in the last section, learning rate and accuracy will be adversely affected. One additional difficulty with this approach is that it fixes the order of the decisions. If some instance is encountered that happens not to have values for the decision that is highest in the tree, then the information in the hierarchy may be inaccessible.

A single set of disjoint categories or a strict hierarchy, when applied in cross-classification domains, will either require excessive storage, extended learning, or will produce less than optimal prediction behavior.

2.2.2 Cross-classification domains

These difficulties are hardly significant if humans only rarely encounter cross-classification domains. On the other hand, if cross-classification domains are common and humans are optimally adapted to their world(Anderson, 1988), then humans must learn overlapping categories.

Linnaeus' taxonomy of the plant and animal kingdom was one of the most significant early indications that hierarchies of disjoint categories reflect the structure of the world. An important source suggesting the existence of overlapping categories in the world is artificial intelligence and cognitive science work in knowledge representation. As structured representations for knowledge appeared, it was acknowledged that the same event or object would need to be multiply classified (Minsky, 1975). Minsky argued that a generator from a car is an instance of both a mechanical system and an electrical one. These different points of view would likely be useful for different tasks. Most proposals for generic knowledge representation languages include the ability to identify multiple classes for an instance (eg. *ConjGeneric* in KRYPTON, Brachman, Fikes, & Levesque, 1983). A particular person can be classified as both a male and a parent and can inherit inferences from each.

The primary reason cited in the representation literature for the use of multiple inheritance is that it limits duplication of information and eliminates unnecessary subclasses (Touretzky, 1986). For example, different animals can have different roles in human life, some are pets, some are circus performers, and some are work animals. A strict hierarchy might require a node for each animal-role pair. With ten different animals and ten different roles, the hierarchy would have at most 100 nodes. The memory requirements could be greatly reduced if circus elephants, for example, could inherit properties from both *circus-performer* and *elephant.* If such overlapping categories were permitted, memory would only require 20 nodes to represent most of the same information.

The world does seem to present cross-classification domains. There are many simple examples of combined concepts, such as pet fish or striped apple (see Osherson & Smith, 1982). Additionally, as mentioned above, researchers concerned with representing knowledge are able to adopt overlapping categories to simplify their task, suggesting that the structure is available. Both of these pieces of evidence argue that because English words express overlapping categories, there must be such structure in the world. Anderson (1990), however, cautions against assuming that category labels correspond to the actual category structure of a domain. The labels may be merely attributes like any other. Two categories may or may not share a particular label. Therefore, the apparent existence of overlapping categories might be the result of some name or label attributes that overlap across categories, just as the extension of *red thing* and the extension of *round thing* overlap.

A category label, however, is assumed to be something special, in that it is extraordinarily useful for predicting other attributes. This is true for the above examples. The category, *circus performer*, indicates much about the lifestyle of the creature, and *elephant* indicates much about the size and shape of the creature. Hence, even if the categories are only labels, there are important predictive relations between the label and other attributes. A strict hierarchy would require that the predictive relations about, for example, circus performers would be duplicated for all types of circus performer. Overlapping categories could permit the clusters of predictive relations to be encapsulated and reused, rather than duplicated. Again, systems of disjoint categories produce inferior performance when compared with systems of overlapping categories.

3 Models of Overlapping Categories

Overlapping categories, by encapsulating predictive relationships, produce more accurate inferences and reduce duplication of information. For this to be true, the various categories must complement each other. In particular, different categories that apply to the same object must differ with respect to the attributes that they are best able to predict. If something is a *beanbag* and a *chair*, the *beanbag* category is best for predicting shape, whereas the *chair* category is best

for predicting size and function.

There are two general approaches to learning sets of complementary, overlapping categories. They be learned either one at a time or simultaneously. Suppose the task is to learn, among other things, the concepts *young* and *mallard*. A learner might at one time hear about a young mallard and only care about predicting body covering and means of locomotion. It would then begin to learn the category, *mallard*. At another time, it might care about predicting degree of coordination or source of food and hence will begin to learn the category *young* for animals. Over time, the two overlapping categories would become more entrenched as young and mallard occurred in other contexts. Both *young* and *mallard* will still participate in sets of disjoint categories. For instance, young things contrasts with old things and mallard contrasts with geese. The resulting structure for the example can then be viewed as an interwoven set of alternate hierarchies.

A simultaneous strategy for learning overlapping categories would produce a similar structure, but would do it with fewer instances. Instead of starting with particular prediction goals, the simultaneous strategy begins with the general goal of extracting as much predictive structure as possible. With each instance, it attempts to maximize its ability to predict all attributes. It would begin to learn about both *young* and *mallard* simultaneously.

3.1 Model 1: Learning overlapping categories across presentation

One possible model for learning overlapping categories across instance presentations uses a single set of categories that at any one time are considered disjoint. An instance, however, can be classified into different categories depending on what the prediction goal is. In this model, an instance is learned by first highlighting which attribute or attributes will be most useful to predict. Then, with this in mind, the instance is incorporated into the best category just as in Fisher's (1987) model. Prediction proceeds in the same manner and results in prediction of the most probable values for the goal attributes.

To actually implement this idea, Fisher's basic algorithm was adopted. However, the Gluck and Corter (1985) measure was inadequate, because it always assumed that all attributes are equally important to predict. Their measure was modified (see Martin & Billman, 1990) to produce the following:

$$Q = \sum_k \sum_l \frac{1}{N_I} P(C_k \mid matchfnc \wedge I_l) \sum_j \sum_i P(a_{A_j}) P(A_j = V_{ij} \mid C_k)^2 \qquad (2)$$

This metric is fairly complicated, but is based directly on Gluck and Corter's simple result. It was modified to allow some attributes to be ignored when learning categories. The term $P(a_{A_j})$ differentiates goal attributes from others. It is 1 if A_j is a goal attribute and 0 otherwise. The term N_I is the number of instances observed, $matchfnc$ is the procedure used for matching, and I is a particular instance that is matched. The match function may be any metric that assigns a degree of match between a particular instance and each disjoint category. Simple possibilities are an arithmetic or geometric average.

3.2 Model 2: Learning overlapping categories simultaneously

The major difference between this model and the last is that this model attempts to pull as much predictive structure out of the instances as it can. It allows simultaneous multiple classification. During prediction, an instance is multiply classified and predictive information is combined from the recognized categories to make a prediction. As an illustration of the idea, imagine that each of several guards has a limited view of the outer wall of a fortress, and that they each report to a

commander, who collects the separate pieces of information. The commander can compose the different fragments to permit informed predictions.

Learning in this model consists of adding new categories, modifying the descriptions of categories, and modifying parameters of the composition function. The system can concurrently learn new categories and learn how to combine the information provided by the categories. If the size of a pet fish is not well predicted from information about pets and fish, either a new pet-fish category would be added or parameters of the composition function would be altered to handle this exception.

More specifically, the model assumes that there are several *non-disjoint* sets of two or more *disjoint* categories. The simplest case is when each set has two possibilities. For example, one category may split plants into those that live underwater and those that do not. Another may split plants into those that are large and those that are small. One category from each the non-disjoint sets could be accessed by an instance. Both learning and prediction use Equation 2 to classify instances into each set in the same manner as in Model 1.

Composition is achieved by assuming that the non-disjoint sets and their activated members are attributes and values of instances that can be categorized by another set of disjoint categories. For each attribute, Equation 2 and the accessed categories are used to select a category from which to generate predictions.

3.3 Empirical illustration

The above models were compared to COBWEB using a cross-classification domain. The domain had twelve binary attributes grouped into four clusters of three mutually predictive attributes. In each cluster, each attribute's values were consistently paired with values on the other two attributes.

Testing was divided into 5 runs for each model. For each run, the domain of 16 instances was randomly ordered and split into a 12 instance training set and a 4 instance test set. After each instance in the training set was presented, the model's ability to make correct predictions about the test instances was determined.

The averaged results demonstrate the predicted benefit of overlapping categories. Both of the overlapping category models achieved a maximum score of approximately 70% correct after all training instances had been seen (67.5% for the first model and 72.5% for the second). COBWEB, on the other hand, achieved only 48.5% correct. One interesting result is that COBWEB only required one instance to achieve a prediction accuracy of $50\% \pm 2\%$. In contrast, the overlapping category models required four and three instances respectively. This suggests that the improved accuracy might be associated with slower initial learning.

4 Discussion

Disjoint categories divide the world at its joints (Rosch, 1978). The different levels of a taxonomy divide the world into either coarse or fine pieces. In these terms, overlapping categories allow the system to identify and consolidate pieces that recur often. Instead of repeatedly defining categories that extract the same piece, that piece is learned once and made generally available.

4.1 Theory-based concept learning

The above statement implies that general predictive rules could be learned and used in a system that permits overlapping categories. These general rules may constitute a theory that can then influence later learning. A simple way this might happen is if previous learning had established a

set of interpredictive values, and future learning assumed that the set of values remained interpredictive. When learning in a new domain, past learning can transfer, possibly permitting faster learning.

Murphy and Medin (1985) argue that general theories do not simply coexist with categories; they interact in many ways. Most importantly, a background theory can provide an explanatory principle for category membership. Although the current representation for theories includes no relationships aside from cooccurrence, conceptual coherence can be simulated with overlapping categories.

There may, for example, already be categories that predict weight from height, and that predict height from gender. Now the learner must learn to partition individuals based on their weight, but they are not given either the weight or the height. All of the information necessary to reason that women are generally lighter than men is available. The background theory permits the identification of important attributes, such as gender, and provides a weak theory to explain why the lighter category contains mostly women.

To fully develop the use of overlapping categories as theories, a more complex representation is needed, including causal and other relations between attributes.

4.2 Conceptual combination

Several overlapping categories can be learned and can be recombined into novel configurations about which the system will have inferential commitments. Someone might, for example, expect a toy elephant to be gray because all elephants they knew were gray and toys have no characteristic color. This general notion is almost identical to the intuition behind conceptual combination (Osherson & Smith, 1982). People are assumed to have separate meaning representations for various nouns and adjectives that can be combined. The major goal of the conceptual combination research is to determine how inferences from separate representations are combined, especially when they are competing. That is also a major issue for models of overlapping categories. The first model presented above makes no commitments about how it might combine two or more categories

The second model, however, adopts a relatively novel perspective on conceptual combination. It does assume a simple multiplicative combination rule, but unlike existing models of combination (Osherson & Smith, 1982), it can modify how the combination is achieved as it learns. So in contrast to earlier approaches, there is not a simple fixed mechanism, but rather a more flexible adaptive method.

4.3 Summary

Although taxonomies have long been the standard model for human categories, they do not always provide the most economical storage nor the best predictions. In particular, in cross classification domains, structures that permit overlapping categories are superior. Two models were presented that learned overlapping category structure, and they were demonstrated to be superior to a method that relies on non-overlapping structure.

5 Acknowledgements

I would like to thank Dorrit Billman for her advice, and I would like to thank Nancy Smith Martin and Tom Hinrichs for assistance with earlier drafts of this paper.

References

Anderson, J. R. (1988). The place of cognitive architectures in a rational analysis. In *Proceedings of Tenth Annual Conference of the Cognitive Science Society*, Montreal, Canada.

Anderson, J. R. (1990). *Adaptive Character of Thought.* Erlbaum, Hillsdale, NJ.

Brachman, R. J., Fikes, R. E., and Levesque, H. J. (1983). Krypton: A functional approach to knowledge acquistion. *IEEE Computer*, 16:67–73.

Bruner, J. S., Goodnow, J. J., and Austin, G. A. (1956). *A Study of Thinking.* Wiley, New York.

Fisher, D. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine Learning.*

Gluck, M. and Corter, J. (1985). Information, uncertainty, and the utility of categories. In *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*, Irvine, CA.

Gluck, M. A. and Bower, G. H. (1988). Evaluating an adaptive network model of human learning. *Journal of Memory and Language*, 27:166–195.

Martin, J. D. (1989). Reducing redundant learning. In *Proceedings of Sixth International Workshop on Machine Learning*, Ithaca, NY.

McClelland, J. L. and Rumelhart, D. E. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol I.* MIT Press, Cambridge, MA.

Medin, D. L. and Shoben, E. J. (1988). Context and structure in conceptual combination. *Cognitive Psychology*, 20:158–190.

Minsky, M. (1975). A framework for representing knowledge. In Winston, P. H., editor, *The Psychology of Computer Vision.* McGraw-Hill, New York.

Murphy, G. L. and Medin, D. L. (1985). The role of theories in conceptual coherence. *Psychological Review*, 92:289–316.

Osherson, D. N. and Smith, E. E. (1982). Gradedness and conceptual combination. *Cognition*, 12:299–318.

Rosch, E., Mervis, C. B., Gray, W. D., Johnson, D. M., and Boyes-Braem, P. (1976). Basic objects in natural categories. *Cognitive Psychology*, 8:382–439.

Touretsky, D. S. (1986). *The Mathematics of Inheritance Systems.* Pitman Publishing, London.

The Right Concept at the Right Time: How Concepts Emerge as Relevant in Response to Context-Dependent Pressures

Melanie Mitchell and Douglas R. Hofstadter
Center for Research on Concepts and Cognition
Indiana University

Abstract

A central question about cognition is how, faced with a situation, one explores possible ways of understanding and responding to it. In particular, how do concepts initially considered irrelevant, or not even considered at all, *become* relevant in response to pressures evoked by the understanding process itself? We describe a model of concepts and high-level perception in which concepts consist of a central region surrounded by a dynamic nondeterministic "halo" of potential associations, in which relevance and degree of association change as processing proceeds. As the representation of a situation is built, associations arise and are considered in a probabilistic fashion according to a *parallel terraced scan*, in which many routes toward understanding the situation are tested in parallel, each at a rate and to a depth reflecting ongoing evaluations of its promise. We describe a computer program that implements this model in the context of analogy-making, and illustrate, using screen dumps from a run, how the program's ability to flexibly bring in appropriate concepts for a given situation emerges from the mechanisms we are proposing.

Suppose you invite your friend Greg to dinner, and he doesn't show up on time. What do you do? At first, simple, standard explanations and actions come to mind: he was briefly delayed; he ran into traffic; he had trouble parking. But as half an hour passes, then an hour, then two, the explanations and actions you think of become more and more out of the ordinary. The following might come to mind: call his office (no answer); call his apartment (no answer); check your calendar to make sure the dinner date *is* tonight (it is); rack your brains trying to remember if he warned you he might be late (no such memory); call friends of his to see if they know where he is (they don't); call his parents in Philadelphia (haven't heard from him in weeks); call the police (they suggest checking the hospital); call the hospital (not there); go to his apartment (not there); ask his neighbors if they've seen him lately (last saw him this morning); drive along routes he would likely have taken (he's nowhere to be seen); buy a megaphone and call out his name as you drive along; call several airlines to see if he's on a plane leaving town tonight; turn on the TV to see if you can spot him sitting in the audience of his favorite talkshow; and so on. Though the last few are outlandish, most of these thoughts *did* occur to the authors when they were in such a situation. The point is: as time goes by and pressure builds up, one's thoughts go farther and farther out on a limb. One considers things one never would have considered initially, letting seemingly unquestionable aspects of the situation "slip" under mounting pressure (e.g., Did I dream I invited him? Did we have a falling-out I forgot about? Did he leave town and not tell me?).

This example illustrates some critical issues in cognition: Faced with a situation, how does one explore possible ways of understanding it, explaining it, or acting in response to it? How do concepts initially considered irrelevant, or not even considered at all, *become* relevant in response to pressure? How does one let go of notions that *looked* relevant but turn out not to be of help after all?

We are studying these issues by developing a model of concepts and high-level perception in which a concept consists of a central region surrounded by a dynamic, probabilistic "halo" of potential associations (Hofstadter, 1988). In its halo, "driving" has such concepts as "parking", "getting stuck in traffic", "having an accident", etc., each with a degree of association that changes in response to context (a phenomenon often discussed by psychologists, e.g., Tversky, 1977; Barsalou, 1989). The halo has no fixed boundary; it cannot be said absolutely that a given concept *is* or *is not* associated with "driving". Instead, different degrees of association reflect probabilities that once a concept is seen as relevant, various associated concepts will also become relevant. The dynamic nature of relevance and conceptual distance imbues human concepts with flexibility and adaptability.

Not only are certain concepts *explicitly* present in one's mental representation of a situation (you consciously believe Greg is *driving*); there are also *implicit* associations with those concepts, most of which stay well below the level of awareness. Given Greg's lateness, the thought that he's driving might easily evoke an image of his having trouble parking (a strong association). However, it is less likely that, early on, you will imagine him in a car accident. This weaker association is *potentially* there, but will not be brought into the picture without pressure (he is quite late, it is dark outside, etc.) This illustrates a general point: far-out ideas (or even ideas slightly past one's defaults) cannot continually occur to people for no good reason; a person to whom this happens is classified as crazy or crackpot. Time and cognitive resources being limited, it is vital to resist nonstandard ways of looking at situations without strong pressure to do so. As an extreme example, had the Michelson–Morley experiment come out the other way (i.e., it had proved there *is* an "ether") and

had Einstein *still* proposed special relativity, with all its deeply counterintuitive notions, it would have been seen as just a fascinating crackpot theory, not a great scientific advance. Not only is pressure needed for one to bring in previously uninvolved concepts in trying to make sense of a situation, but the concepts brought in are related to the *source* of the pressure; they are a function of the pressure. (These ideas overlap with Kahneman & Miller's 1986 treatment of counterfactuals.)

One aim in our model is to avoid two opposite strategies, both psychologically implausible, for searching through concepts to be used in understanding a given situation: (1) All concepts are *explicitly* and *equally* available from the start (e.g., you have a preconstructed list of concepts relevant to "late-dinner-guest" situations — you may not need to try them all out, since Number 4 on the list might fill the bill, but they are spelled out nonetheless. An equally implausible variant of this would be that the possibilities are not spelled out explicitly, but it is easy to generate the next one on the list if a given entry fails); and (2) Certain concepts are definitively excluded from the start, and can *never* be brought in as relevant. A premise of our model is that in humans, the presence or absence of a concept in a situation is not black-and-white; rather, all one's concepts should have the *potential* to become relevant in any situation, but due to the necessity for cognitive economy, they can't all be made available all the time or to the same degree. People resist even *generating* less standard views, not to mention *exploring* them; the less standard a view, the more it is resisted.

In our model, every concept possessed by the system has *some* probability of becoming relevant in every context, but different concepts have vastly different probabilities, and these vary with context. There are many possible explanations (you could have written down the wrong date or given Greg the wrong address; your street's name could even have been secretly changed), so it is important not to *absolutely* exclude any particular pathway ahead of time. All must be potentially open, but there is not enough time to explore all equally, or even to generate all. Allocation of cognitive resources to different pathways must be a *dynamic* function of context-dependent pressures, because those pressures might change as exploration proceeds (when you try to call Greg, you find your phone is out of order and no one can call in; this will tend to make the "car accident" pathway less plausible). Our model proposes that many potential pathways are being tested out all the time, but at different speeds and to different levels of depth: due to context-dependent pressures, not all pathways are tested equally. Some may not be considered at all, but that's the luck of the (biased) draw; the point is that they are *potentially* open for exploration. We term this non-egalitarian style of exploration a *parallel terraced scan*: many different pathways are explored in parallel, but not equally; each pathway is explored at a rate and to a depth proportional to moment-to-moment estimates of its promise.

Our model thus has two interrelated aspects: The first is the existence of a probabilistic halo of potential associations around the central region of each concept. Like an electron orbit in an atom, a concept is blurry and distributed in "semantic space", with various probabilities that it will "be" at a given spot. For concepts, as for electrons, the probability distribution changes in a context-dependent way. The seond aspect is the notion of a parallel terraced scan. These ideas are implemented in Copycat, a computer model of concepts and high-level perception in analogy-making.

To isolate many issues of general psychological import, we use an idealized microworld in which these issues emerge very clearly. Our methodology resembles that of physics, where problems are *idealized* in order to isolate what is interesting about them and to allow them to be studied more precisely. In this spirit, Copycat operates in a "frictionless" world consisting of analogy problems involving letter strings; despite their apparent simplicity, these problems capture many of the broad issues we are investigating. Four sample problems in Copycat's microworld are:

1. **abc** ⇒ **abd**; **ijk** ⇒ ?
2. **abc** ⇒ **abd**; **iijjkk** ⇒ ?
3. **abc** ⇒ **abd**; **kji** ⇒ ?
4. **abc** ⇒ **abd**; **xyz** ⇒ ?

Solving such problems requires many abilities necessary for high-level perception and analogy-making in general: mentally building a coherently structured whole from initially unconnected parts; describing objects, relations, and events at an "appropriate" level of abstraction; paying attention to relevant aspects and ignoring irrelevant and superficial aspects of situations; deciding which elements of a situation to chunk and which to view individually; deciding which descriptions to take literally and which to let slip when perceiving correspondences between aspects of two situations; and allowing competition among various ways of interpreting and mapping the situations. Discussions of how problems 1–4 require these abilities and how Copycat solves 1–4 are given in Hofstadter & Mitchell (1988) and Mitchell & Hofstadter (1990). Our goal is not to study the domain-specific mechanisms people use in solving letter-string analogies, but to develop a computer model of human flexibility and insight in general; we use this microworld because it cleanly isolates many of the abilities we are investigating.

The central issue of this paper — how dormant concepts "bubble up" in response to pressure and become relevant — arises somewhat in problems 1–4, but is manifested most clearly in this one:

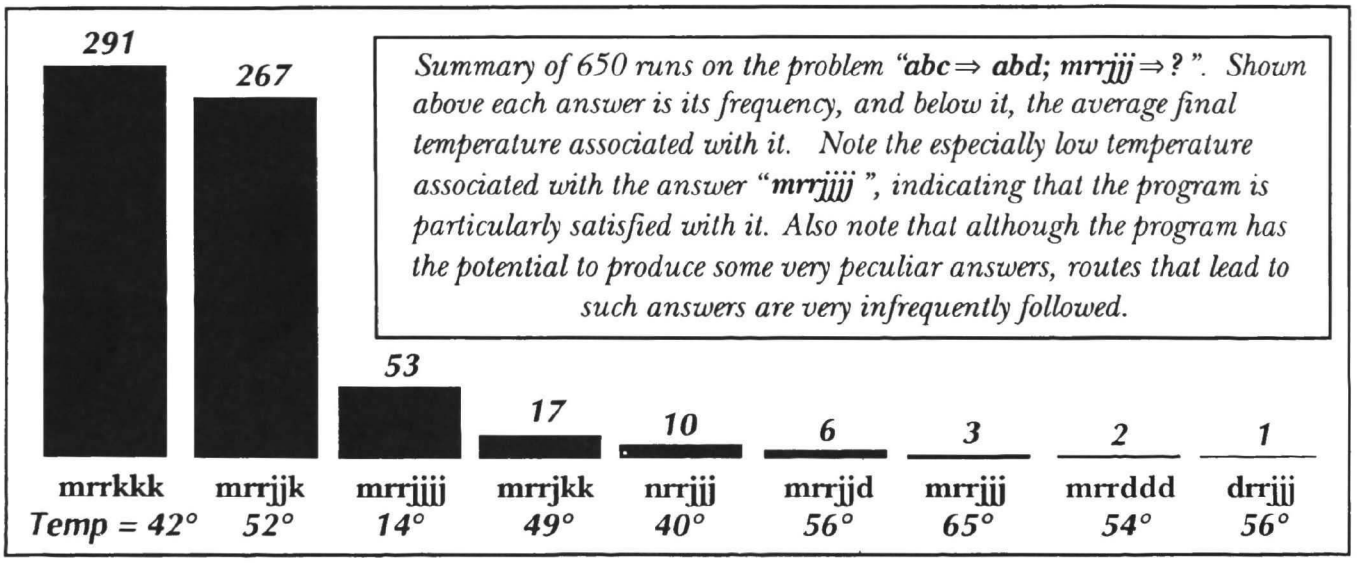
5. **abc ⇒ abd; mrrjjj ⇒ ?**

This problem has a seemingly reasonable, straightforward solution: **mrrkkk**. Most people give this answer, reasoning that since **abc**'s rightmost letter was replaced by its successor, and since **mrrjjj**'s rightmost "letter" is actually a *group* of 'j's, one should replace all the 'j's by 'k's. Another possibility is to take "rightmost letter" literally, thus to replace only the rightmost 'j' by 'k', giving **mrrjjk**. However, neither answer is very satisfying, since neither takes into account the salient fact that **abc** is an alphabetically increasing sequence. This internal "fabric" of **abc** is a very appealing and seemingly *explanatory* aspect of the string, so you want to use it in making the analogy, but how? No such fabric seems to weave **mrrjjj** together. So either (like most people) you settle for **mrrkkk** (or possibly **mrrjjk**), or you look more deeply. But where to look when there are so many possibilities?

The interest of this problem is that there happens to be an aspect of **mrrjjj** lurking beneath the surface that, once recognized, yields what many people feel is a more satisfying answer. If you ignore the *letters* in **mrrjjj** and look instead at *group lengths*, the desired successorship fabric is found: the lengths of groups increase as "1-2-3". Once this hidden connection between **abc** and **mrrjjj** is discovered, the rule describing **abc ⇒ abd** can be adapted to **mrrjjj** as "Replace the length of the rightmost group by its successor", yielding "1-2-4" at the abstract level, or, more concretely, **mrrjjjj**. Thus this problem demonstrates how a previously irrelevant, unnoticed aspect of a situation emerges as relevant in response to pressures (e.g., the unsatisfied desire for a common fabric, among others).

How can the notion of group length, which in most problems remains essentially dormant, come to be seen as relevant by Copycat? *Length* is certainly in the halo of the concept *group*, as are concepts such as *letter-category* (e.g., 'j' for the group 'jjj'), *string-position* (e.g., *rightmost*), and *group-fabric* (e.g., *sameness between letters*). Some are more closely associated with *group* than others; in the absence of pressure, the notion of *length* tends to be fairly far away in conceptual space. Thus in perceiving a group such as 'rr', one is virtually certain to notice the letter-category ('r'), but not very likely to notice, or at least attach importance to, the length. However, since *length* is in *group*'s halo, there is *some* possibility that lengths will be noticed and used in trying to make sense of the problem. One might consciously notice a group's length at some point, but if this doesn't turn out to be useful, *length*'s relevance diminishes after a while. (For example, this might happen in the variant problem **abc ⇒ abd, mrrrrjj ⇒ ?**.) This *dynamic* aspect of relevance is very important: even if a new concept is at some point brought in as relevant, it is counterproductive to continue spending much of one's time exploring avenues involving that concept if none seems promising.

Since Copycat is nondeterministic, it follows different paths on different runs; thus not only does it come up with a variety of answers, but it can reach each answer in myriad ways. Indeed, Copycat's flexibility depends on the fact that all pathways involving any of its concepts are potentially open; despite this, the program generally manages to avoid exploring unpromising pathways, except fleetingly. Below is a chart showing the results of running Copycat some 650 times on problem 5. Its answers, ordered by frequency, range from the superficially alluring **mrrkkk** to the downright bizarre **mrrjkk** (in which the two rightmost 'j's were perceived as a chunk), **nrrjjj** (in which **abc**'s *rightmost* letter was equated with **mrrjjj**'s *leftmost* letter), **mrrjjj** (using the rule "Replace all 'c's by 'd's"), and **drrjjj**.



*Summary of 650 runs on the problem "**abc ⇒ abd; mrrjjj ⇒ ?**". Shown above each answer is its frequency, and below it, the average final temperature associated with it. Note the especially low temperature associated with the answer "**mrrjjjj**", indicating that the program is particularly satisfied with it. Also note that although the program has the potential to produce some very peculiar answers, routes that lead to such answers are very infrequently followed.*

Although there are only nine distinct answers, each run was unique on a fine-grained level. Under each answer is the *final temperature* averaged over all runs yielding that answer. Temperature is explained later; for the time being, think of a run's final temperature as a measure of Copycat's "happiness" with the answer produced, with high temperature corresponding to low happiness and vice versa. Thus Copycat is by far the happiest with **mrrjjjj**. Note the lack of correlation of frequency with final temperature — meaning, roughly, that obviousness and elegance are independent.

The frequencies shown in the chart are not meant to be strictly compared with the frequencies of various answers given by people to this problem, since, as we said earlier, the program is not meant to model the domain-specific mechanisms people use in solving these letter-string problems. Rather, what is interesting here is that the program *does* have the potential to arrive at very strange answers (such as **mrrjkk** and **drrjjj**), yet manages to steer clear of them almost all the time; most always, it gets answers that people find reasonable and, sometimes, even insightful.

In a complex world (even one with the limited complexity of Copycat's microworld), one never knows in advance what concepts may turn out relevant in a given situation. It is thus imperative not only to avoid dogmatically *open-minded* search strategies, which entertain all possibilities equally seriously, but also to avoid dogmatically *closed-minded* search strategies, which in an ironclad way rule out certain possibilities *a priori*. Copycat opts for a middle way, which of course leaves open the potential for disaster — and indeed, disaster occurs once in a while. This is the price that must be paid for flexibility. People, too, occasionally explore and even favor peculiar routes. Our program, like us, has to have the potential to concoct far-out solutions in order to be able to discover subtle and elegant ones like **mrrjjjj**. (In fact, Copycat still lacks important mechanisms that would allow it to pursue yet stranger pathways!) To rigidly close off any routes *a priori* would necessarily remove critical aspects of Copycat's flexibility. On the other hand, the fact that Copycat so rarely produces weird answers demonstrates that its mechanisms manage to strike a pretty effective balance between open-mindedness and closed-mindedness, imbuing it with both flexibility and robustness.

We now sketch one way Copycat arrives at **mrrjjjj** (more details given below). The input consists of three "raw" strings (here, **abc**, **abd**, **mrrjjj**) with no preattached relations or preformed groups; it is thus left entirely to the program to build up perceptual structures constituting its understanding of the problem in terms of concepts it deems relevant. On most runs, the groups 'rr' and 'jjj' are constructed (the program is able to perceive copy-groups — groups consisting of repeated copies of a given letter — quite readily). Each group's letter-category ('r' and 'j' respectively) is explicitly noted, since *letter-category* is relevant by default. There is some probability for lengths to be noticed at the time the groups are made, but it is low, since *length* is not strongly associated with *group*. Once 'rr' and 'jjj' are made, *copy-group* becomes very relevant. This creates top-down pressure for the system to describe other objects — especially in the same string — as copy-groups if possible. The only way to do this here is to describe the 'm' as a copy-group with just one letter. This is strongly resisted by an opposing pressure: a single-letter group is an intrinsically weak and far-fetched construct. However, the existence of two other copy-groups in the string, coupled with the system's "unhappiness" at its failure to incorporate the lone 'm' into any large, coherent structure, pushes against this resistance.

These opposing pressures fight; the outcome is decided probabilistically. If the 'm' is perceived as a single-letter group, its length will very likely be noticed (single-letter groups are noteworthy precisely because of their abnormal length), making *length* more relevant in general, and thus increasing the probability of noticing the other two groups' lengths. Moreover, *length*, once brought into the picture, has a good chance of staying relevant, since descriptions based on it turn out to be useful. (Note that had the string been **mrrrrjj**, *length* might be brought in, but it would not turn out useful, so it would likely fade back into obscurity.) In **mrrjjj**, once lengths are noticed, the successor relations among them are quickly constructed by relation-detectors continually seeking new relations. There is also an independent top-down pressure to see successor relations in **mrrjjj**, coming from the already-seen successor relations in **abc**. As this satisfying new view of **mrrjjj** begins to emerge, the importance of the groups' letter-categories fades and *length* becomes their most salient aspect. Thus the crux of discovering this solution lies in the triggering of the concept *length*.

In summary, Copycat's solution of **abc** ⇒ **abd**, **mrrjjj** ⇒ **mrrjjjj** requires the interaction of:

- *concepts consisting of a central region surrounded by a halo of potential associations;*
- *a mechanism for probabilistically bringing in new associations related to the current situation;*
- *a mechanism by which concepts' activations decay over time, unless reinforced;*
- *agents that continually seek new relations, groups, and correspondences;*
- *mechanisms for applying top-down pressures from concepts already brought in;*
- *mechanisms allowing competition among pressures;*
- *the parallel terraced scan, allowing rival views to develop at different speeds.*

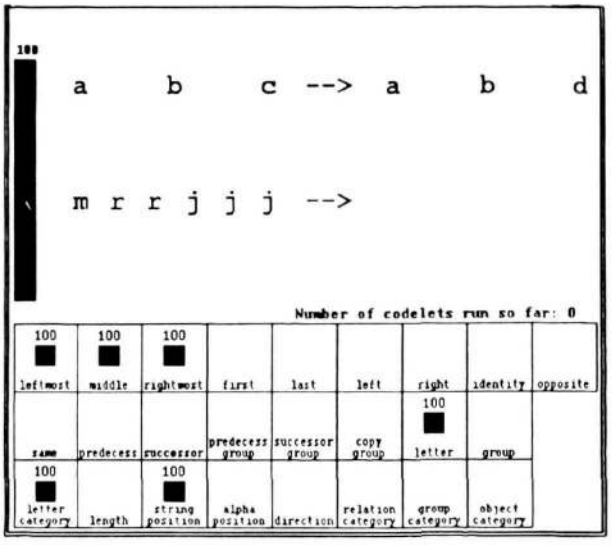
We now describe and illustrate how these mechanisms are implemented. (For more details, see Mitchell & Hofstadter, 1990.) Copycat's concepts reside in a network of nodes and links called the *Slipnet*. A concept's central region is a node, and its associative halo corresponds to other nodes linked to the central node. A node (such as *copy-group* or *successor*) becomes activated when instances of it are perceived (by *codelets*, as described below), and loses activation unless its instances remain salient. A node spreads activation to nearby nodes as a function of their proximity. Activation levels are not binary, but can vary continuously. The probability a node will be brought in or be considered further at any given time as a possible organizing concept is a function of the node's current activation level. Thus there is no black-and-white answer to the question of whether a given concept is "present" at a given time; continuous activation levels and probabilities allow different concepts to be present to different degrees. All concepts have the potential to be brought in and used; which ones become relevant and to what degree depends on the situation the program is facing, as will be seen below.

In addition to the Slipnet, where long-term concepts reside, Copycat has a working area in which perceptual structures (e.g., descriptions, relations, groups, and correspondences) are built hierarchically on top of the "raw" input (the three letter-strings). This building process is carried out by large numbers of simple agents called *codelets*. A codelet is a small piece of code that carries out some small, local task that is part of the process of building a structure (e.g., one codelet might notice that the two 'r's in **mrrjjj** are the same letter; another codelet might estimate how well that proposed relation fits in with already-existing relations; another codelet might build the relation). *Bottom-up* codelets work toward building structures based on whatever they happen to find, without being prompted to look for instances of specific concepts; *top-down* codelets look for instances of particular active nodes, such as *successor* or *copy-group*. The probability at a given time that a node in the Slipnet will add a top-down codelet to the current codelet population is a function of the node's activation level. Any structure is built by a series of codelets running in turn, each deciding probabilistically on the basis of its estimation of some aspect of the structure's strength whether to continue, by generating one or more follow-up codelets, or to abandon the effort at that point. If the decision is made to continue, the running codelet assigns an *urgency* value (based on its estimate of the structure's promise) to each follow-up codelet. This value helps to determine how long each follow-up codelet will have to wait before it can run and continue the evaluation of that particular structure.

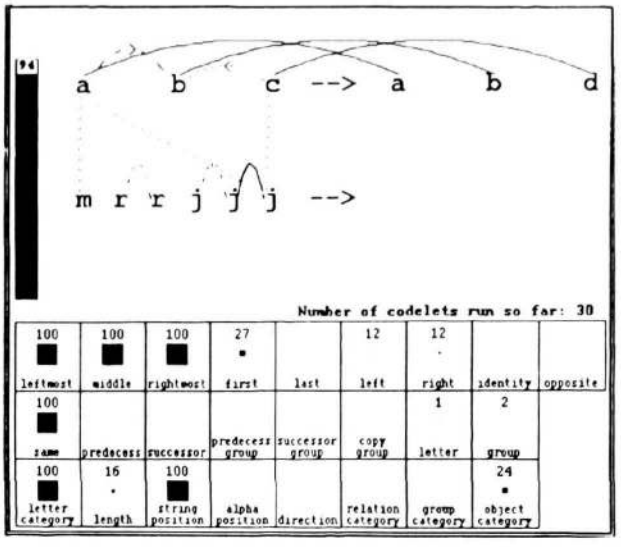
Any run starts with a standard initial population of bottom-up codelets with preset urgencies; at each time step, one codelet is chosen to run and is removed from the current codelet population. The choice is probabilistic, based on relative urgencies in the current population. As the run proceeds, new codelets are added to the population either as follow-ups to previously-run codelets, or as top-down scouts for active nodes. A new codelet's urgency is assigned by its creator as a function of the estimated promise of the task it is to work on. Thus the codelet population changes as the run proceeds, in response to the system's needs as judged by previously-run codelets and by activation patterns in the Slipnet, which themselves depend on what structures are built. There is no top-level executive directing the system's activity; all processing is carried out by codelets.

The fine-grained breakup of structure-building processes serves two purposes: (1) it allows many such processes to be carried out in parallel, by having their components interleaved; and (2) it allows the computational resources allocated to each such process to be dynamically regulated by moment-to-moment estimates of the promise of the pathway being followed. A process is not a predetermined macroscopic act that is then broken up into convenient chunks; rather, any sequence of codelets that amounts to a coherent macroscopic act can *a posteriori* be labeled a process — thus processes are *emergent*. The speed of any process emerges dynamically from the urgencies of its component codelets. The upshot is a parallel terraced scan — more promising views tend to be explored faster than less promising ones.

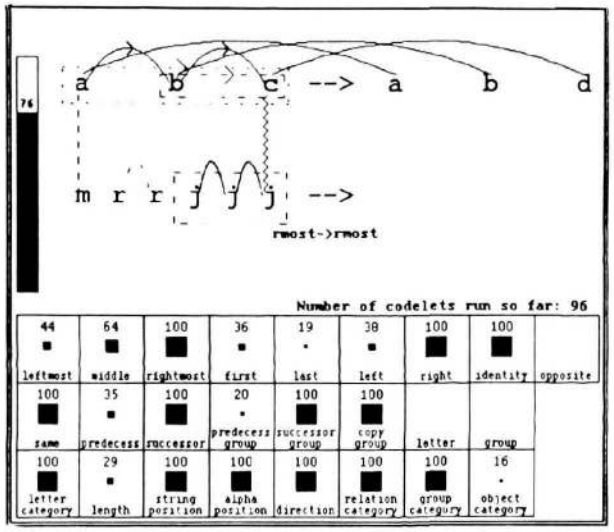
A final mechanism, *temperature*, both measures the degree of perceptual disorganization in the system (its value at any moment is a function of the amount and quality of structure built so far), and controls the degree of randomness used in making decisions (e.g., which codelet should run next, which structure should win a competition, etc.). Higher temperatures reflect the fact that there is little information on which to base decisions; lower temperatures reflect the fact that there is greater certainty about the basis for decisions. Temperature in Copycat is described in detail in Mitchell & Hofstadter (1990). All these mechanisms are illustrated in the set of screen dumps from a run of the program, given below.



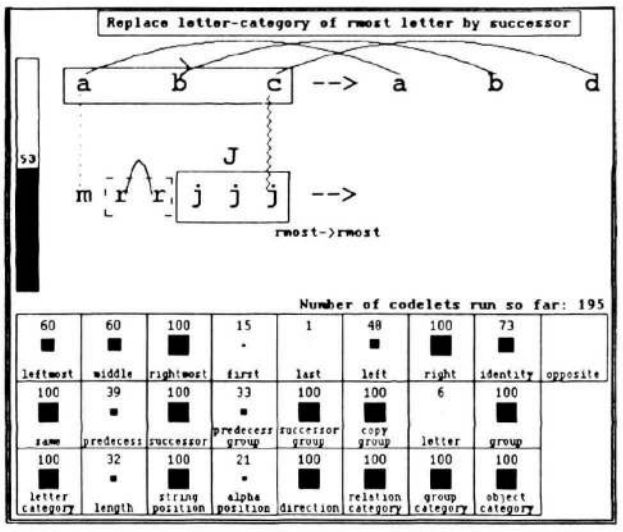
1. The problem is presented. Temperature, shown on a "thermometer" (left), is at its maximum of 100 (no structures yet built). At the bottom, some Slipnet nodes are displayed (links not shown). (Due to limited space, many nodes are not shown, e.g., those for 'a', 'b', etc.) A black square represents a node's current activation level (the actual value, from 0 to 100, is shown above). Nodes here displayed include *string-positions* of objects (*leftmost, middle,* and *rightmost*); *alphabetic-positions* of letters (*first,* filled by 'a', and *last,* by 'z'); *directions* for relations and groups (*left* and *right*); *identity* and *opposite* (some of the possible relations between concepts); *relation-categories* for relations between letters and groups (*same, predecessor,* and *successor*); *group-categories* (*predecessor-group, successor-group,* and *copy-group*); *object-categories* (*letter* and *group*); and in row 3, nodes representing these various categories of descriptions, including *length*. Every letter has some preattached descriptions: *letter-category* (e.g., 'm'), *object-category* (*letter,* as opposed to *group*) and *string-position* (*leftmost, middle, rightmost,* or none — e.g., the fourth letter in **mrrjjj** has no string-position description). These nodes start out highly activated.



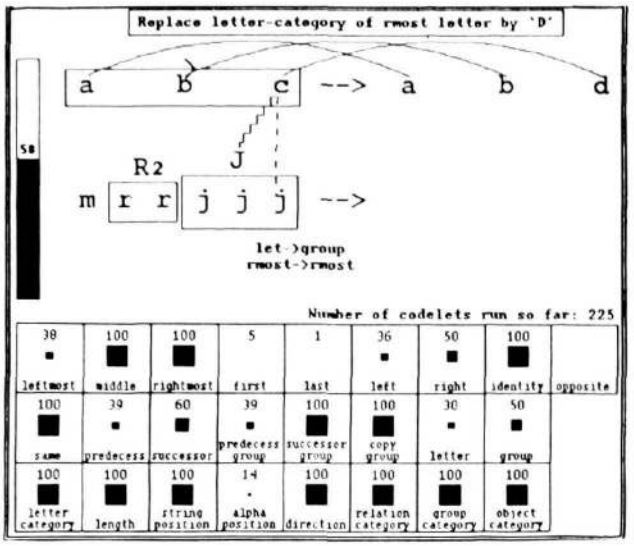
2. The 30 codelets so far run have begun exploring many possible structures. Dashed lines and arcs are structures in various stages of consideration; solid ones are structures actually *built,* which can thus influence temperature, and the building of other structures. Relations and correspondences between letters are being considered (the 'a'–'j' potential mapping is based on the weak *leftmost–rightmost* Slipnet link; being implausible, it won't be pursued much further). The **abc/abd** correspondences and the rightmost 'j'–'j' sameness relation have been built by bottom-up codelets; this activated *same,* resulting in top-down pressure (new codelets) to seek sameness elsewhere. Some nodes have become lightly activated via spreading activation (e.g., the node *first,* from 'a' [not shown]). *Length*'s activation comes from its weak association with *letter-category* (letters and numbers are increasing sequences and thus similar; numbers are associated with length). Temperature has fallen in response to structures so far built. Many non-displayed fleeting explorations are occurring (e.g., "Any relation between the 'm' and its neighbor 'r'?").



3. The successorship fabric of **abc** has been seen, and two mutually competing groups based on it are being considered: 'bc' and 'abc'. The latter is much stronger than the former, thus has a much higher chance. Exploration of the diagonal 'a'–'j' correspondence was aborted. A 'c'–'j' correspondence has been built (jagged vertical line); its reason for existence (both letters are rightmost) is given beneath it. A 'jjj' group is being strongly considered. Since *successor* and *sameness* relations have been built, these nodes are highly active; they in turn have spread activation to *successor-group* and *copy-group,* which creates top-down pressure to look for such groups. Also, since *first* was active, *alphabetic-position* became highly active (a probabilistic event), making alphabetic-position descriptions likely to be considered.

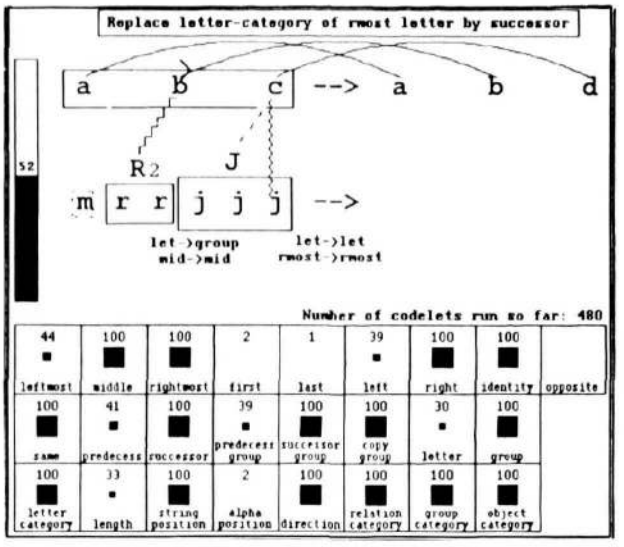


4. Groups 'abc' and 'jjj' have been built (relations between letters are no longer being displayed). An 'rr' group is being considered (the group 'jjj' strongly supports it, so its construction is accelerated). Meanwhile, a *rule* (at top) has been constructed to describe how **abc** changed. The current version of Copycat assumes the example change involves replacing exactly one letter, so rule-building codelets fill in the template "Replace ____ by ____", choosing probabilistically from descriptions the program has given to the changed letter and its replacement, with a default bias toward more abstract descriptions (e.g., usually preferring "rightmost letter" to 'c'). Nodes *first* and *alphabetic-position* didn't turn out useful and thus have faded. Also, *length* received additional activation from *group* but is still not very activated, so noticing lengths is still unlikely.

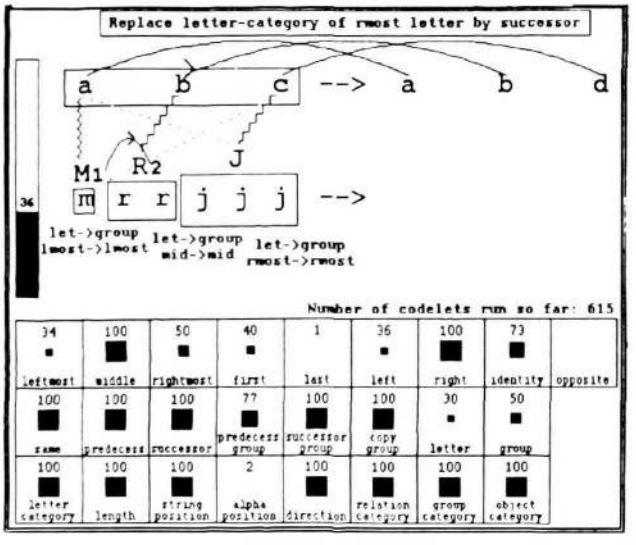


5. Now, 225 codelets into the run, the letter-to-*letter* 'c'–'j' correspondence was defeated by a stronger letter-to-*group* 'c'–'J' correspondence, though the former possibility still lurks in the background. Meanwhile, an 'rr' group was built whose length was noticed (a probabilistic event) and is displayed at the top of the group. *Length* is now fully active.

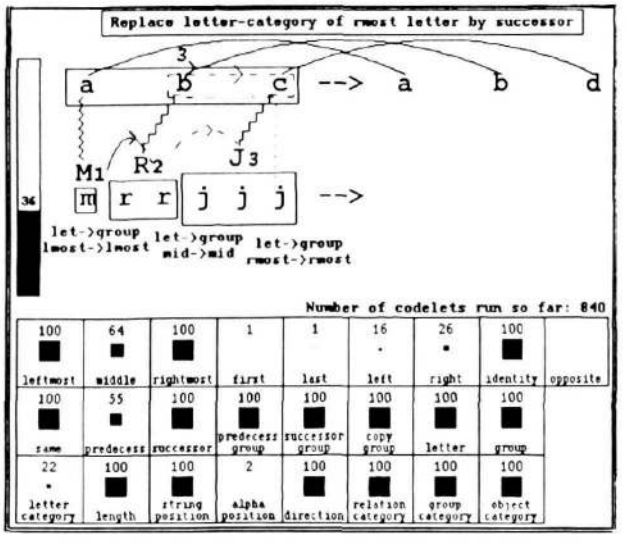
A new rule, "Replace the letter-category of the rightmost letter by 'D'", appears at the top of the screen; although it is weaker than the old rule, fights are decided probabilistically, and it won. However, its weakness caused temperature to go up. If the program were to stop now (unlikely, as temperature is still fairly high; the program decides probabilistically when to stop, based on temperature), the rule would be adapted for **mrrjjj** as "Replace the letter-category of the rightmost *group* by 'D'" (the 'c'–'J' correspondence establishes that the role of *letter* in **abc** is played by *group* in **mrrjjj**), yielding **mrrddd** (and Copycat does get this answer on occasion).



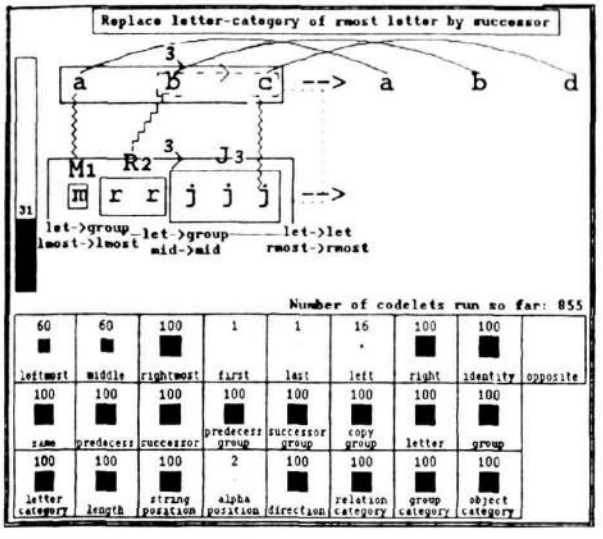
6. The previous, stronger rule has been restored (again the result of a fight having a probabilistic outcome), but the 'c'–'J' correspondence has been defeated by a 'c'–'j' correspondence. The activation of *length* has decayed a good deal, since the length description given to 'rr' hasn't been found to be useful. (This is graphically indicated by the fact that the '2' is no longer in boldface.) The temperature is still fairly high, since the program is having a hard time making a single, coherent structure out of **mrrjjj**, as it did with **abc**. That fact, combined with strong top-down pressure from the two copy-groups in **mrrjjj**, makes it somewhat plausible for the system to flirt with the idea of a single-letter group (dashed rectangle around the 'm').



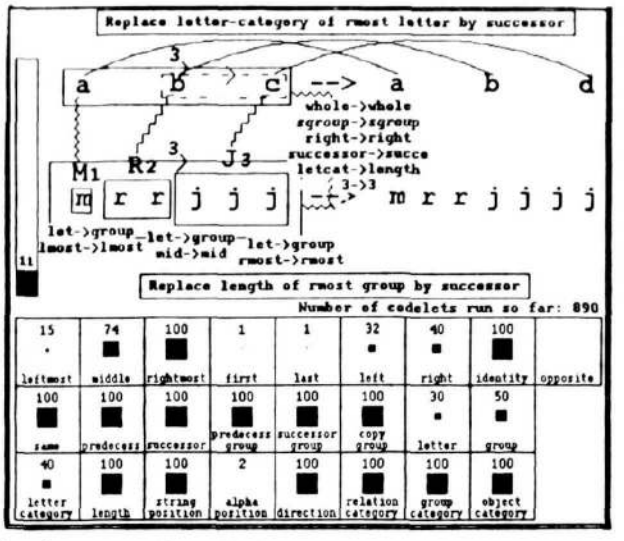
7. As a result of these pressures, the *a priori* extremely unlikely single-letter copy-group 'm' has been built, and its length of 1, being very noteworthy, has been attached as a description. The relation between the 1 and the 2 has been built; all of this is helping *length* to stay active. A complete set of *letter* ⇒ *group* correspondences has now been made, and as a result of these promising new structures, the temperature has fallen to 36, which in return helps to lock in this emerging view.



8. As a result of *length*'s continued activity, length descriptions have been attached to the other two groups in the problem ('jjj' and 'abc'), and a relation between the 2 and the 3 (for which there is much top-down pressure coming from both **abc** and the emerging view of **mrrjjj**) is being considered. *Letter-category* has decayed, indicating that it hasn't lately been of use in building structures.



9. The 2–3 relation was built and a successor-group was built out of the group-lengths in **mrrjjj** (large rectangle surrounding the three copy-groups). Also, a correspondence (dashed vertical line to the right of the two strings) is being considered between **abc** and **mrrjjj** in their entireties.



10. A correspondence has been built between the two strings as wholes, and its concept-mappings (e.g., *letter-category* ⇒ *length*) are listed to its right. The original rule has been translated, using these concept-mappings; the translated rule appears just above the Slipnet, and the answer **mrrjjjj** at the right. The low temperature reflects the program's satisfaction with this answer.

In summary, Copycat is a model of concepts and perceptual mechanisms flexible enough to deal with a range of problems in its microdomain, reflecting many central psychological issues. It differs from other computer approaches to analogy-making in that it models not only how situations are mapped onto each other, but also the mechanisms by which initially uninterpreted situations are mentally structured. Models such as SME (Falkenhainer et al., 1986), and ACME (Holyoak & Thagard, 1989), are concerned with just the mapping process; all relations and other perceptual structures are precoded into predicate-logic representations that serve as input. A detailed comparison of Copycat with these models is given in Mitchell (1988). Copycat starts on any problem from a standard initial state; however, it quickly senses unique aspects of the problem, bringing out certain associations while downplaying others, allowing it (usually) to home in on a suitable set of relevant concepts and avenues of approach. Copycat achieves, through mechanisms we believe are psychologically plausible, a delicate balance between being too open-minded (exploring every avenue indiscriminately, thus grossly wasting computational resources) and being too closed-minded (rigidly cutting out certain avenues *a priori*, thus preventing many creative pathways from ever being looked at). Walking this fine line imbues Copycat with its robustness and flexibility.

Acknowledgments

We thank Robert French for contributions to the Copycat Project, and Liane Gabora for writing the statistics-gathering program. Thanks also to David Chalmers and David Moser for helpful comments on this paper, and to Greg Huber for a "late" inspiration. This research has been supported by grants from Indiana University, the University of Michigan, and Apple Computer, Inc., as well as a grant from Mitchell Kapor, Ellen Poss, and the Lotus Development Corporation, and grant DCR 8410409 from the National Science Foundation.

References

[1] Barsalou, L. W. (1989). Intraconcept similarity and its implications for interconcept similarity. In Vosniadou, S. and A. Ortony, *Similarity and analogical reasoning*, 76–121. Cambridge, England: Cambridge University Press.

[2] Falkenhainer, B., K. D. Forbus, and D. Gentner (1986). The Structure-Mapping Engine. In *Proceedings of the American Association for Artificial Intelligence, AAAI-86*. Los Altos, CA: Morgan Kaufmann.

[3] Hofstadter, D. R. (1988). Common sense and conceptual halos. *Behavioral and Brain Sciences, 11* (1), 35–37.

[4] Hofstadter, D. R. and M. Mitchell (1988). Conceptual slippage and analogy-making: A report on the Copycat project. *Proceedings, Tenth Annual Cognitive Science Society Conference*. Hillsdale, NJ: Lawrence Erlbaum Associates.

[5] Holyoak, K. and P. Thagard (1989). Analogical mapping by constraint satisfaction. *Cognitive Science, 13* (3), 295–355.

[6] Kahneman, D. and D. T. Miller (1986). Norm theory: Comparing reality to its alternatives. *Psychological Review, 93* (2), 136–153.

[7] Mitchell, M. (1988). A computer model of analogical thought. Unpublished thesis proposal. University of Michigan, Ann Arbor, MI.

[8] Mitchell, M. and D. R. Hofstadter (1990). The emergence of understanding in a computer model of concepts and analogy-making. *Physica D*, in press.

[9] Tversky, A. (1977). Features of similarity. *Psychological Review, 84*, 327–352.

Classification of Dot Patterns with Competitive Chunking

Emile Servan-Schreiber
Department of Psychology, Carnegie Mellon University

Abstract

Chunking, a familiar idea in cognitive science, has recently been formalized by Servan-Schreiber and Anderson (in press) into a theory of perception and learning, and it successfully simulated the human acquisition of an artificial grammar through the simple memorization of exemplar sentences. In this article I briefly present the theory, called Competitive Chunking, or CC, as it has been extended to deal with the task of encoding random dot patterns. I explain how CC can be applied to the classic task of classifying such patterns into multiple categories, and report a successful simulation of data collected by Knapp and Anderson (1984). The tentative conclusion is that people seem to process dot patterns and artificial grammars in the same way, and that chunking is an important part of that process.

Introduction

Chunking, our natural tendency to process stimuli by parts, is one of the most familiar and powerful ideas of cognitive science. Strangely, apart from Laird, Newell, and Rosenbloom's recent *Soar* theory (Newell, in press), there have been no serious attempts to formalize chunking since its discovery by Miller 34 years ago (Miller, 1956). Recently, Servan-Schreiber and Anderson (in press) have demonstrated how a chunking program can simulate human subjects learning an artificial grammar through mere memorization of exemplar sentences. That program was based on a theory called *competitive chunking*, or *CC*. In this paper, I demonstrate how *CC* can be applied to the classic problem of classifying random dot patterns.

Competitive Chunking of Dot Patterns

Representation: what is a chunk?

A chunk is a long term memory hierarchical structure whose constituents are chunks also. Every chunk has an associated *strength* which is a composite score reflecting how often and recently it has been used in the past. A newly created chunk has a strength of one unit. Its strength is increased by an additional unit every time it is used, or re-created. Strength also decays with time. At any point in time, the strength of a chunk is the sum of its successive, individually decaying, strengthenings:

$$\text{Strength} = \sum_i T_i^{-d} \qquad (1)$$

where T_i is the time elapsed since the ith strengthening, and d, the decay parameter, determines the severity of strength decay ($0 < d < 1$). Once a chunk is created, it exists for ever, and there is

no limit on how much strength it can accumulate. This strength construct is identical to that of ACT* for declarative memory traces (Anderson, 1983).

When the stimuli are dot patterns, two kinds of chunks are assumed: dot-chunks, and complex-chunks. Dot-chunks encode a single stimulus dot, whereas complex-chunks encode a pair of chunks. For example, the dot-chunk ((35 22)) encodes a dot located at cartesian coordinates (35 22) on the stimulus matrix. Examples of complex chunks are: (((35 22)) ((100 75))) which encodes two dot-chunks, and ((((35 22)) ((100 75))) ((125 200))) which encodes a complex-chunk and a dot-chunk.

Processes: What are chunks for?

Chunks are used to perceive stimuli. In the case of dot patterns, a percept is a collection of stimulus dots and chunks. Perception consists of multiple passes through a percept elaboration cycle, where the elementary (pre-elaboration) percept is simply the set of stimulus dots present in the pattern: chunks are retrieved as competing candidates to build upon the current percept, and some are selected. The selected chunks are used to elaborate on the percept, yielding a new percept as a basis for another cycle. The cycle repeats until no chunks are retrieved for further elaboration.

To be retrieved as a candidate for elaboration, a chunk must *match* some part of the current percept. Matching is defined slightly differently for dot-chunks and complex-chunks. A dot-chunk has a certain probability of matching any dot that is present within its immediate surroundings. This probability is 1 for the dot that is encoded by the dot-chunk, and decreases exponentially with the distance between the encoded dot and the target dot:

$$\text{Probability of match} = e^{-m \cdot \text{distance}} \qquad (2)$$

where m, the match parameter, determines how steep the exponential is ($m > 0$). The larger m is, the harder it is for a dot-chunk to match distant dots. Figure 1 plots this function for $m = .05$.

A complex-chunk matches some part of the percept if and only if both of its subchunks are equivalent to chunks in the percept. Chunk equivalence is defined recursively: The equivalence between two dot-chunks is probabilistic, depending on the distance between their encoded dots, following Equation (2). Then, two complex-chunks are equivalent if and only if (a) they have the same hierarchical structure, and (b) they have equivalent terminal dot-chunks at the bases of their hierarchies.

Still, it isn't enough that a chunk matches some part of the percept for it to be retrieved as a candidate for elaboration. The match must also have enough *support*. The support for a complex-chunk match is defined as the summed strength of the matched chunks. But the support for a dot-chunk match is infinite. The probability that a chunk match is retrieved is then a negatively accelerated increasing function of its support:

$$\text{Probability of retrieval} = \frac{1 - e^{-c\,\text{support}}}{1 + e^{-c\,\text{support}}} \qquad (3)$$

where c, the competition parameter, determines the steepness of the probability curve ($c > 0$). The larger c is, the easier it is to retrieve chunks, at all levels of support. Figure 2 plots this function for $c = .5$.

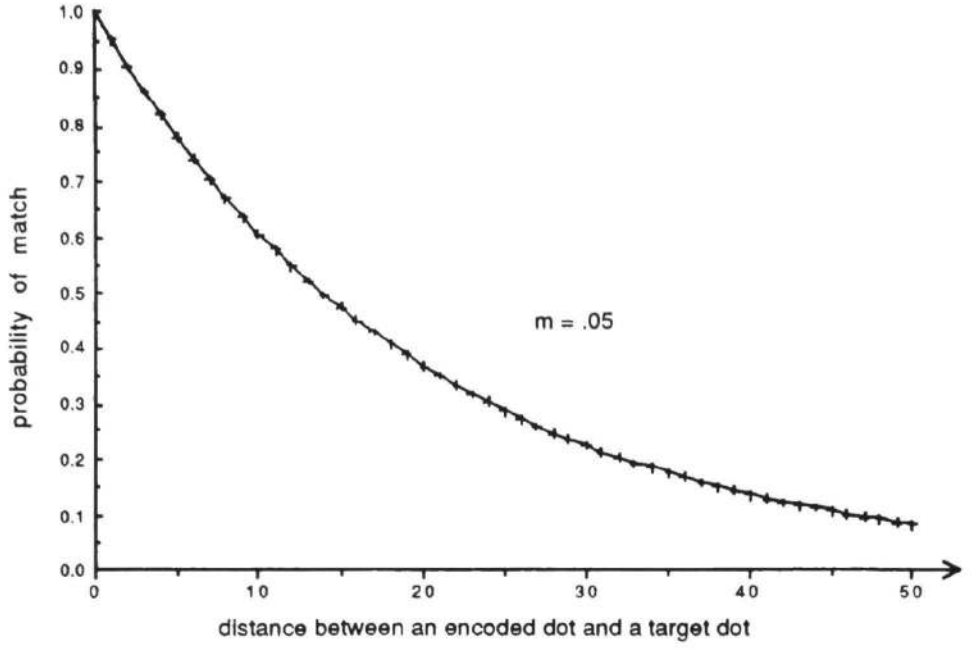


Figure 1. Plot of the probability that a dot-chunk matches a stimulus (target) dot. It decreases exponentially as the distance between the dot that is encoded by the dot-chunk and the target dot increases, following Equation (2). The value of m that is shown is the one I used in the simulation I describe later.

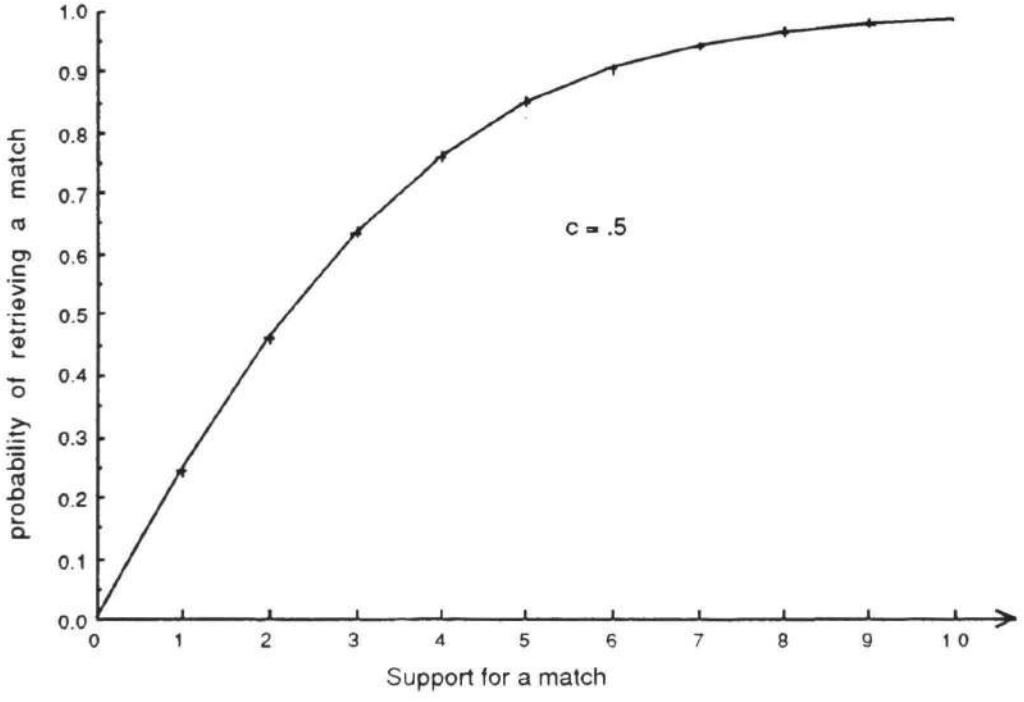


Figure 2. Plot of the probability that a chunk match is retrieved as a candidate for elaboration. It increases as its support increases, following Equation (3). The value of c that is shown is the one I used in the simulation I describe later.

Once chunk matches have been retrieved, they are organized into sets of compatible matches. Two matches are compatible if they match different parts of the percept. The alternative sets then compete against each other for the privilege of elaborating on the percept. For each set, the sum of the strengths of its constituent chunks is computed, and the set of matches with the highest score is selected as the winner. Elaboration then consist of replacing the parts of the percept that were matched by the chunks in the selected set of matches.[1] Every elaborating chunk is strengthened, while their losing competitors are left to decay. Note that this selection process, based on compatibility and strength, favorises sets of many matches over sets of fewer matches. It also encourages the participation of weaker chunks as long as they are compatible with a number of stronger chunks. This, in turn, allows them to gain strength.

It is interesting to note the dual role that strength plays in the elaboration process: The probability that a matching chunk is retrieved depends on the strength of the chunks that it matches , its support, while the probability that it is selected among the retrieved chunks depends on its own strength. Thus, a chunk's strength is a critical parameter for both itself and the chunks that may match it. When a chunk is strengthened, both itself and the chunks that may match it are being learned. Conversely, when a chunk's strength is left to decay, both itself and the chunks that may match it are being forgotten. A chunk is being learned the fastest, then, when its subchunks tend to be equivalent to chunks that co-occur frequently in the environment.

Stimulus Familiarity.

Given that chunks are, by definition, familiar units of knowledge, it follows that the more chunks participate in the percept elaboration process, the more familiar the stimulus is perceived to be. The notion of familiarity can easily be formalized in CC. Note that every time that a complex-chunk elaborates on the current percept, two of the percept's chunks are replaced by a single chunk in the next percept. Thus, elaborating on the percept with complex-chunks has the effect of reducing the number of parts at the top level of the percept. The implication is that the more chunks participated in the elaboration process, the less parts there are in the final percept. Therefore, the relationship between familiarity and the number of parts in the final percept, call it nchunks, can be characterized as follows: the larger nchunks is, the less familiar the stimulus is perceived to be. Conversely, the smaller nchunks is (its minimum value is 1), the more familiar the stimulus is perceived to be. To formalize further, Servan-Schreiber and Anderson (in press) assumed that familiarity can take values from 1 (maximum) to an asymptotic 0 (minimum), and that it is a rapidly decreasing function of nchunks:

$$\text{Familiarity of stimulus} = e^{1 - \text{nchunks}} \qquad (4)$$

This formula captures the notion that if a stimulus can be sufficiently elaborated upon so that the final percept consists of a single chunk, then it is perceived as maximally familiar.

---

[1] Making the elaboration process set-based is a departure from its earlier specification in Servan-Schreiber and Anderson (in press). In that earlier verion of the theory, matches competed individually for elaboration, and a single match, that with the strongest chunk, was selected at each cycle.

Chunk Creation.

Learning in CC is two-fold. As discussed above, existing chunks are learned when they are strengthened. Strengthening and strength decay allow for the tuning of the existing knowledge base. But CC also has a process for chunk creation.

The creation process is a direct extension of the perception process, and shares many of its characteristics. The input to the creation process is the final percept, and its output is a collection of new chunks. The goal of that process is to create chunks that will have a good chance of participating in the perception process should the same or a similar stimulus be presented, thus increasing its familiarity by reducing nchunks. To that end, the proposed new chunks are those that would have enough support to be retrieved. Because stimulus dots provide infinite support to dot-chunks that encode them, if the final percept contains some stimulus dots that were not matched by any dot-chunk, then new dot-chunks are created to encode them. If the final percept contains two or more chunks, then a new complex-chunk is proposed that encodes the pair of chunks, in the percept, with the largest summed strength. Because that measure is akin to the proposed new chunk's support, Equation (3) is used to compute the probability that it is created. A newly created chunks is given a strength of one unit. If it already exists then it is simply strengthened. CC does not keep multiple copies of a chunk.

Applying CC to the Classification Task

General Principles.

A classic design of dot-pattern classification experiments includes a training phase and a testing phase. In the training phase, subjects are shown distortions of three prototype patterns and are instructed to classify them into three categories. When they make an incorrect classification, they are given feedback on the correct response. In the testing phase, the feedback is suppressed and the patterns that must be classified are of at least three kinds: distortions of the prototypes that were shown during the training (OLD), distortions of the prototypes that were not shown during training (NEW), and the prototypes themselves (PRO). The dependent variables of interest are then the percentages of correct classifications of each kind of pattern.

When CC is presented with a dot pattern, it builds as compact a percept as it can. The output of the perception process is then nchunks, a measure of how familiar the stimulus is perceived to be. When its task is to classify, CC keeps multiple separate sets of chunks, one per category. Then, when a pattern is presented, CC can compute multiple values of nchunks, one per chunk set, and select the category that is associated with the set of chunks that yielded the smallest value of nchunks, the most familiar percept. If feedback on the correct classification is given, as in the training phase, then it can be used to guide the creation of new chunks. Chunks are created only from the percept associated with the correct category, and the new chunks become part of the set of chunks associated with that category. The next time that a pattern from that category is presented, CC has thus increased its chance of building a compact percept with the chunks from

the correct category.

To make the selection of a response, on each trial, more dependent on the context provided by the previous trials, I decided to transform the values of nchunks into relative familiarity scores, or f-scores. Given a particular set of chunks, an *f-score* is simply the ratio of the average value of nchunks for all previous stimuli to the value of nchunks for the current stimulus. Therefore the smaller the value of nchunks for the current stimulus is, compared to its average value for previous stimuli, the larger its f-score is. An f-score that is less than 1 indicates that the current stimulus appears less familiar than previous stimuli have (on average). An f-score that is larger than 1 indicates the converse. The response selection rule is then to select the category that is associated with the set of chunks that yields the largest f-score.

Experimental Test of the Theory.

To test CC's ability to classify random dot patterns, I selected the experimental design of Knapp and Anderson (1984). There are 3 categories. During training, a single distortion of category A's prototype is presented 24 times, 6 distortions of category B's prototype are presented 4 times each, and 24 distortions of category C's prototype are presented once each, for a total of 72 training patterns, 24 per category. The 3 prototypes are generated by placing 9 dots at random locations in a 300 by 300 array, and the distortions are generated by moving each dot in a prototype exactly 25 array units from its original location, in a randomly chosen direction. (Three prototypes are randomly generated for each subject.) At test, OLD, NEW, and PRO patterns are presented from each category, 8 patterns representing each of the 9 possible combinations of pattern kind and category, yielding 72 testing trials.

Knapp and Anderson found (a) that the correct classification of OLD patterns decreased as the number of different exemplars seen during training increased, (b) that the correct classification of NEW and PRO patterns, on the contrary, increased with the number of different exemplars seen during training, and (c) that the PRO patterns were always more correctly classified than the NEW patterns.

There are three parameters to be set in CC: the decay parameter d, the competition parameter, c, and the match parameter, m. Due to the high computational cost of running simulated subjects, I did not try many different combinations of values for these parameters, but, rather, relied on past experience with CC to select reasonable values. The values of c and d that Servan-Schreiber and Anderson (in press) found appropriate in simulating the acquisition of an artificial grammar were .5 and .5. I used those. For m, which is a new parameter for CC, I relied on Knapp and Anderson's (1984) experience with their own theory that contained a dot matching function very similar to Equation (2). Their experience points to a value of m of about .05. I used that. To reduce the computational cost further, without sacrificing psychological plausibility, CC was allowed to create new chunks only on those training trials when it made an incorrect classification. On testing trials, the chunk creation process was completely turned off, although the strengthening and strength decay processes continued to operate. Time was increased by one unit with every trial.

Figure 3 plots the average classification performance of 50 simulated subjects. Clearly, the qualitative pattern of results reported by Knapp and Anderson (1984) is reproduced. As the number of different exemplars of a category seen during training increases, the classification of OLD patterns suffers, while that of NEW and PRO patterns is enhanced. At the same time, the PRO patterns are always more easily classified than the NEW patterns.

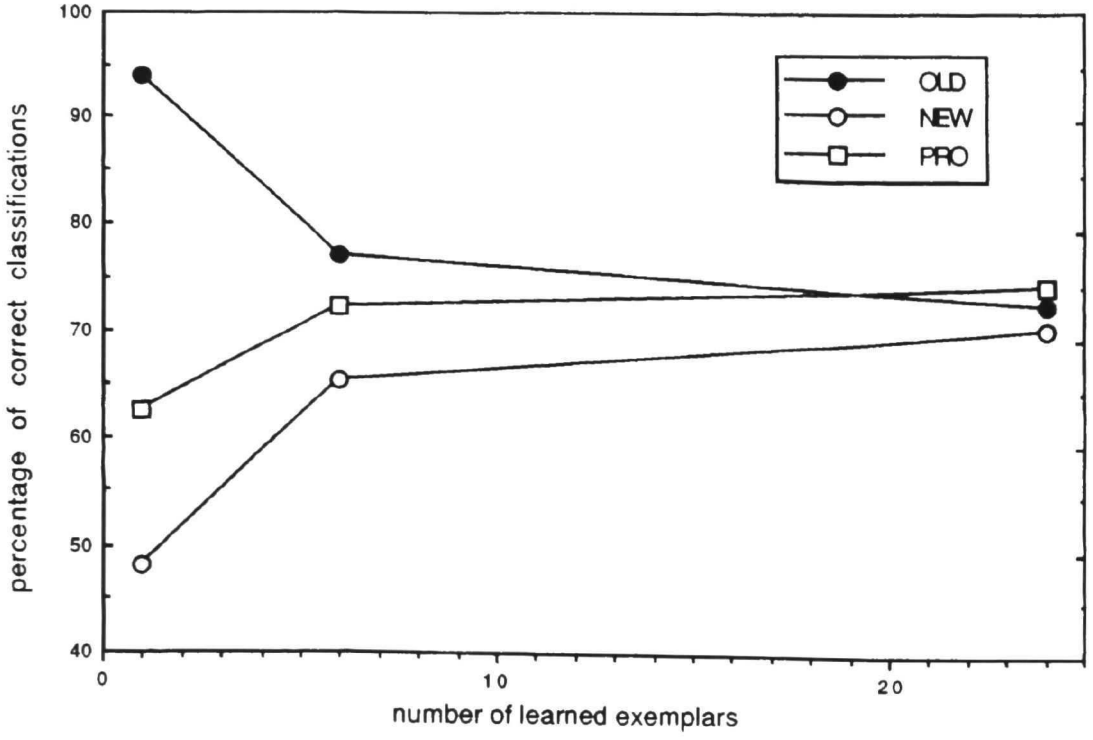


Figure 3. Percentages of correct classifications of OLD, NEW, and PRO patterns in each of the three categories A (1 learned exemplar), B (6 learned exemplars), and C (24 learned exemplars). The values of CC's parameters c, d, and m, in this experiment, were .5, .5, and .05 respectively.

Conclusion

CC already offers a precise and comprehensive theory of how subjects acquire artificial grammars, in the laboratory, through the simple memorization of exemplar sentences. The research I report here is CC's first foray into the classic problem of abstracting visual categories from exemplars. The results of a limited experiment are encouraging, and call for more experimentation with the theory. There is also independent evidence that the classification of dot patterns is likely a fertile ground for a theory of perception and learning based on chunking. Hock, Tromley, and Polmann (1988) report that, in the process of encoding dot patterns, people are very sensitive to configurational cues encoded into what they call "perceptual units" Evidently a synonym for "chunk". CC has the potential to provide a unified explanation of how people abstract category information, through the chunking of exemplars, in both verbal and visual domains.

Acknowledgments

This research was supported by contract N00014-86-K-0678 from the Office of Naval Research. I thank John Anderson for his intellectual support and encouragement. Correspondence regarding this article should be addressed to Emile Servan-Schreiber, Department of Psychology, Carnegie Mellon University, Pittsburgh, Pennsylvania, 15213-3890.

References

Anderson, J. R. (1983). *The Architecture of Cognition.* Cambridge, MA: Harvard University Press.

Knapp, A. G., & Anderson, J. A. (1984). Theory of categorization based on distributed memory storage. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 10*, 616-637.

Hock, H. S., Tromley, C., & Polmann, L. (1988). Perceptual units in the acquisition of visual categories. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 14*, 75-84.

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review, 63*, 81-97.

Newell, A. (in press). *Unified Theories of Cognition.* Cambridge, MA: Harvard University Press.

Servan-Schreiber, E., & Anderson, J. R. (in press). Learning artificial grammars with competitive chunking. *Journal of Experimental Psychology: Learning, Memory, and Cognition.*

Can Causal Induction Be Reduced to Associative Learning?

Michael R. Waldmann
University of Frankfurt

Keith J. Holyoak
University of California, Los Angeles

ABSTRACT

A number of researchers have recently claimed that higher-order human learning, such as categorization and causal induction, can be explained by the same principles as govern lower-order learning, such as classical conditioning in animals. An alternative view is that people often impose abstract causal models on observations, rather than simply associating inputs with outputs. We report three experiments using a multiple-cue learning paradigm in which models based on associative learning versus abstract causal models make opposing predictions. We show that different causal models can yield radically different learning from identical observations. In particular, we compared people's abilities to learn when the positive cases were defined by a linear cue-combination rule versus a rule involving a within-category correlation between cues. The linear structure was more readily learned when the cues were interpreted as possible causes of an effect to be predicted, whereas the correlated structure was more readily learned when the cues were interpreted as the effects of a cause to be diagnosed. The results disconfirm all associative models of causal induction in which inputs are associated with outputs without regard for causal directionality.

Introduction

The Associative View of Multiple-Cue Learning

Tasks as different as classification learning, causal induction, and classical conditioning can be viewed as examples of multiple-cue learning. In each of these tasks, a number of cues, which might be features, causes, or conditional stimuli, are combined to trigger a response. This response might be a classification decision, a prediction of an effect, or a conditioned response. Because of the apparent similarity between different types of multiple-cue learning situations, it is tempting to postulate common underlying learning mechanisms for them. A currently popular view of multiple-cue learning treats it as a bottom-up process that is fundamentally associative in nature. Thus higher-order types of learning in humans, such as classification learning, and lower-order types of learning in animals, such as classical conditioning, are seen as examples of similar learning processes.

A number of researchers have recently claimed that higher-order types of human learning, such as categorization and causal induction, can be explained by principles that govern lower-order learning in animals, such as classical conditioning (e.g., Gluck & Bower, 1988a, b; Shanks & Dickinson, 1987). In particular, Gluck and Bower (1988a, b) have suggested that adaptive associative networks can provide powerful models of human categorization as well as of classical conditioning. These connectionist models consist of an input layer that represents potential cues, such as symptoms of possible diseases observed in a patient, and an output layer that might represent classification responses, such as diagnoses of alternative diseases. The responses are computed by a linear function of the weighted cues. The weights are learned in a competitive fashion using the least mean squares (LMS) learning rule (Widrow & Hoff, 1960), in which the weights are incrementally updated in proportion to the response error they produce. Gluck and Bower have shown that a simple model of this sort compares favorably with other models of human categorization (also see Estes, Campbell, Hatsopoulos, & Hurwitz, 1989; for a critique see Shanks, 1990). Since the LMS rule is formally equivalent to

This research was supported by NSF Grant BNS 87-10305 to Patricia Cheng. Michael Waldmann was sponsored by a grant from the German Research Foundation. We thank Patricia Cheng for helpful advice and discussions.

Rescorla and Wagner's (1972) theory of classical conditioning (Sutton & Barto, 1981), these findings suggest that important commonalities link higher-order learning such as categorization and lower-order classical conditioning. Thus Gluck and Bower basically claim that categorization can be modelled as simple associative learning. Similarly, Shanks and Dickinson (1987) argue that causal induction can be reduced to associative learning.

As pointed out by Minsky and Papert (1969), simple one-layer networks can only learn linearly-separable learning tasks. To deal with this major limitation, various extensions of associative network models have been suggested in the connectionist literature. Gluck, Hee, and Bower (1989) proposed a configural-cue network in which pairwise conjunctions of simple cues are coded using configural cues added to the input layer (see also Gluck & Bower, 1988b). Alternatively, the standard connectionist approach to nonlinear learning tasks is to add intermediate layers of hidden nodes between the input and the output layers, which can be used to code cue combinations (Rumelhart, Hinton, & Williams, 1986). Using backpropagation of error signals, which conceptually is an extension of the LMS learning rule to multiple-layer networks, these hidden units can be trained to code interactions in the input. Despite the differences among the various alternative network models, each of these connectionist learning schemes shares a fundamental associationistic assumption: The network simply tries to learn statistical associations between the nodes coded on the input level and the desired output.

Learning Within Abstract Causal Models

The associative view of learning can be contrasted with a more mentalistic approach, which can be traced back to Gestalt psychology. In this tradition, it is claimed that people use abstract, meaningful world knowledge to guide their learning about new domains. Higher-order learning and lower-order associative learning are seen as different in important ways. In particular, one view of human learning is that people impose abstract causal models on observations. Wattenmaker, Dewey, Murphy, and Medin (1986) have shown that people profit from specific world knowledge. People become more sensitive to structural relations between the input cues during learning when they can relate the learning material to previously acquired knowledge. We will argue here that even in situations in which people cannot bring to bear specific world knowledge, they nonetheless might use abstract knowledge about central properties of the world -- in particular, abstract knowledge about causal relations. We have set up an experimental situation in which associative learning and learning based on abstract causal models can be pitted against each other. We will show that different causal models can yield radically different learning from identical observations, a finding that cannot be explained by associative learning models.

Figure 1 illustrates how we decouple higher-order causal learning from associative learning in our experiments. The arrows represent temporal precedence, either in order of presentation of the information, or in order of cause and effect.

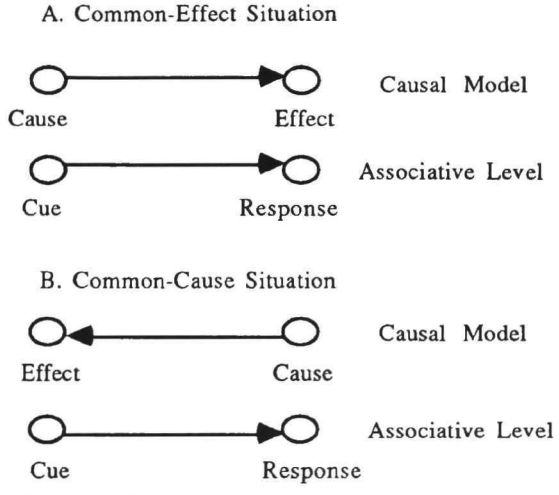


Figure 1. (A) Common-effect situation, in which causal cues are used to predict a potential effect; (B) Common-cause situation, in which presented effects serve as cues to diagnose a potential cause.

The lower halves of both Figure 1A and B show an associationistic representation of the learning situations: Cues are presented first, and the task is to learn to associate them with the correct responses. The corresponding upper halves show how the tasks map to a causal account. In a "common-effect" situation (Figure 1A), the cues represent causes and the responses represent decisions about a predicted effect. Since according to our world knowledge causes always precede effects, the temporal orderings are isomorphic between the causal and associationistic representations of the task. In a "common-cause" situation (Figure 1B), the cues represent effects, and the responses represent decisions about a cause to be diagnosed. The temporal ordering is reversed relative to a common-effect situation, and the mapping to the associationistic description is also reversed. By comparing learning in these two types of situations, this design allows us to disentangle predictions based on associative accounts from those derived from assumptions about causal models for induction. In both tasks, cues have to be associated with the required responses. Thus if subjects treat both tasks as associative multiple-cue learning, then both should yield identical patterns of learning. If, however, subjects represent the two situations in terms of causal models, the two tasks will differ in psychologically important ways, and the learning patterns should reflect these differences.

Experiment 1

Method

A multiple-cue learning task was used in this experiment. Subjects were handed index cards one at a time, with each giving a description of a fictitious person. Subjects were asked to give a "yes-no" response, classifying the cards either as positive or as negative cases. Immediately after every response subjects were told if their judgment was correct or incorrrect. The subjects were trained until they reached a learning criterion (two cycles through the eight basic cases without error) or until they received an upper limit of learning trials.

The descriptions on the index cards consisted of three binary values of dimensional features: weight, pallor, and perspiration. The fictitious persons had either high (e.g., anorexic) or low (e.g., underweight) intensity values on each of these dimensions. The eight possible cases were arranged either in a linearly separable or in a non-linearly separable, correlated fashion (see Figure 2). Similar structures have previously been investigated by Wattenmaker et al. (1986) and by Shepard, Hovland, and Jenkins (1961).

	+				-		
	Case	Dimensions			Case	Dimensions		
		1	2	3		1	2	3
Linearly Separable	1.	H	H	H	5.	H	L	L
	2.	H	H	L	6.	L	L	H
	3.	H	L	H	7.	L	H	L
	4.	L	H	H	8.	L	L	L
Correlated	1.	H	H	H	5.	L	H	H
	2.	H	L	H	6.	H	L	L
	3.	L	H	L	7.	L	L	H
	4.	L	L	L	8.	H	H	L

Figure 2: Structure of item sets used in Experiment 1

The positive set corresponds to a correct "yes" response, and the negative set to a correct "no" response. In the linearly separable arrangement high values of the dimensions are more typical for the positive set, and low values for the negative set. For both sets, each dimension has one exceptional value so that the dimensional values are only probabilistically related to the sets. However, a simple

linear rule distinguishes the two sets. If a person has at least two out of three high values on the three dimensions, then this person belongs to the positive set. This structure does not require hidden layers or configural nodes in a connectionist learning network.

In the correlated, non-linearly separable condition, neither high nor low values are more or less typical for the positive or negative set. For each dimension, there are two persons with high values and two with low values in each set. There is therefore no linear rule to separate the two sets. The only way to distinguish the two sets is to notice the positive correlation between the first and the third dimension in the positive set, and the negative correlation in the negative set. The middle dimension is irrelevant for the classification. This task, which is formally equivalent to learning an "exclusive-or" structure, requires configural nodes or hidden layers in connectionist networks.

This linear-separability factor was crossed with a second factor involving manipulation of the causal structure imposed on the learning task. In the "common-cause" condition, subjects were told that they are going to learn about a disease that is caused by a virus, which could be more or less intense. In this condition the virus plays the role of a common cause that simultaneously affects the symptoms. The cues that subjects saw on the index cards thus correspond to effects of a common cause. This causal model naturally predicts a "spurious correlation" between the effects: A high-intensity virus should yield high-intensity effects, whereas a low-intensity virus should yield low-intensity effects. This situation in fact corresponds to the correlated condition; accordingly, we predicted that this condition should be particularly easy for subjects who received a cover story consistent with the common-cause model.

In a second causal context, the "common-effect" condition, the causal directions were reversed. Now the subjects were told that an experiment on social cognition had been conducted. In this experiment it was found that the appearance of some people produces a new emotional response in their observers. Here the cues on the index cards correspond to potential causes of a common effect. The subjects' task was to learn to predict which person elicits an emotional response in an observer. This emotional response might vary in intensity. Common-effect structures do not imply correlations among the causes. Learning correlated causes amounts to learning a disordinal interaction, whereas the linear condition corresponds to a causal model with three main effects. Given the preference people have for linear as opposed to configural causal structures, the linearly separable task should be relatively easy to learn (see Dawes, 1982).

It is important to note that although subjects were informed that the cause (common-cause condition) or effect (common-effect condition) could vary in intensity, no feedback about the intensity level of the outcome factor was ever provided. Rather, subjects were only told whether the outcome was obtained, regardless of its intensity.

To summarize, if subjects learn according to the accounts of associative learning theories (e.g., connectionist models with hidden layers or configural nodes), the different causal structures imposed on the task should not matter. Subjects across the two causal conditions see identical cues, and are required to learn identical cue-response mappings. However, if subjects are sensitive to the different structural implications of the two causal models, their learning rates for the linear and correlated condition should vary across the two causal cover stories.

Results

Figure 3 shows the results based on 40 UCLA undergraduates who served as subjects. The mean number of errors made prior to the subject reaching the learning criterion was used as an indicator of learning difficulty. As predicted, the causal cover story interacted with the structure of the item set, $F(1, 36) = 7.48$, $p < .025$. The correlated condition was easier to learn in the disease context, in which a correlation naturally falls out of a common-cause structure. In contrast, in the emotional-response condition the linearly-separable item set was easier to learn than the correlated set, as would be expected if people find main-effect models simpler to learn than causal interactions. Overall, the linear condition was learned with fewer errors than was the correlated condition, $F(1,36) = 5.78$, $p < .025$. The results of Experiment 1 thus clearly support the claim that subjects were using causal models during learning, rather than simply trying to associate the presented cues with the correct responses.

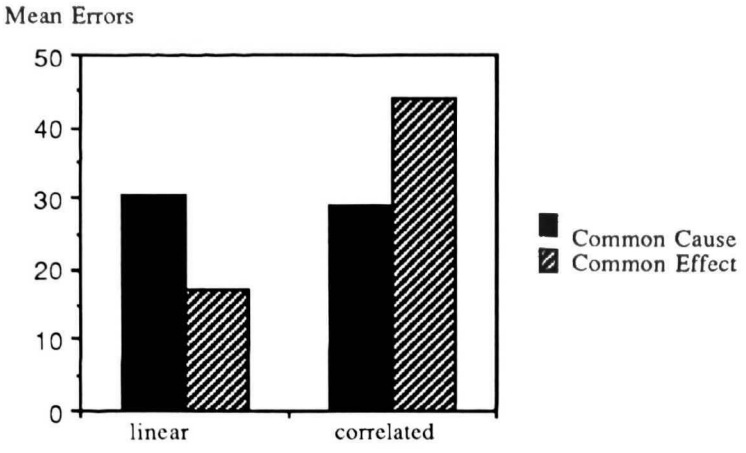


Figure 3. Mean errors prior to reaching criterion as a function of the causal model (common cause vs. common effect) and structure of the item set (linearly separable vs. correlated) in Experiment 1.

Experiment 2

Method

In a second experiment we focused on the correlated condition. In order to better approximate correlations with continuous variables, we used two variables with four intensity levels each in this experiment. We again used weight and pallor as dimensions. The levels of weight were "slightly underweight", "underweight", "seriously underweight", and "anorexic body"; analogous levels were used for pallor. In the positive set these two variables were perfectly positively correlated (values 4 4, 3 3, 2 2, and 1 1, for the four positive items), whereas in the negative set they were negatively correlated (values 4 1, 3 2, 2 3, and 1 4). Note that models like the configural-cue model of Gluck et al. (1989), which introduce separate configural cues for each pairwise feature-value combination, do not capture the monotonicity involved in a correlation of continuous variables. In addition, the number of configural cues required by such models grows exponentially with the number of levels.

In addition to examining learning with more clearly continuous variables, Experiment 2 addressed the question of whether subjects really need explicit information about the fact that the virus (the common cause) may vary in intensity. Even though capturing the positive correlation within the positive set requires the assumption of a continuous common cause, subjects might be able to infer this property of the cause by observing the learning patterns. If the effects are clearly continuous (as was the case for our materials), this may encourage the assumption that the underlying cause is also continuous. Accordingly, half of the subjects received the hint that the common cause might vary in intensity, as in Experiment 1, whereas the other half did not. This hint factor was crossed with the causal context factor, which again consisted of a common-cause and a common-effect condition. Ten subjects served in each of the four conditions.

Results

The results, displayed in Figure 4, replicated the finding that the correlated item set is learned more readily in the common-effect than in the common-cause condition, $F(1,36) = 7.38$, $p < .025$. Omitting the hint that the cause (virus) could vary in intensity did not significantly impair subjects' performance. The impact of causal models on learning correlated item sets thus generalizes to more continuous dimensions.

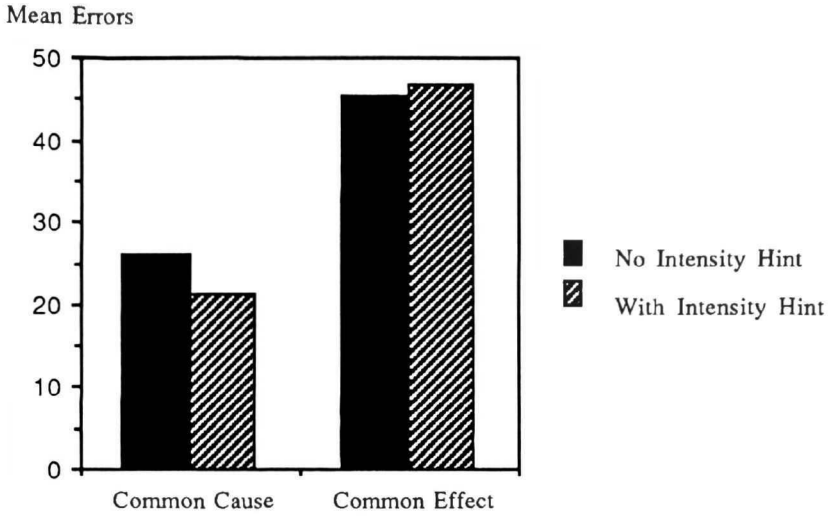


Figure 4. Mean errors prior to reaching criterion as a function of the causal model (common cause vs. common effect) and provision of an intensity hint for a correlated item set in Experiment 2.

Experiment 3

Method

Experiment 3 addresses a restriction that Gluck et al. (1989) imposed on their configural-cue network model. The major advantage of configural-cue networks is that they can learn interactions using simple linear networks with the standard LMS-rule, without requiring backpropagation. Their major problem is that the potential number of configural cues grows exponentially with the number of input cues. Gluck and Bower (1988b) therefore suggest restricting configural cues to pairwise conjunctions. An obvious drawback of this restriction is that such a network is unable to handle problems for which the correct decision requires learning an interaction among three (or more) cues.

	+					-				
Case	Dimensions				Case	Dimensions				
	1	2	3	4		1	2	3	4	
1.	H	H	H	H	5.	H	H	L	H	(L)
2.	H	H	H	L	6.	H	L	H	L	(H)
3.	L	L	L	H	7.	L	H	H	H	(L)
4.	L	L	L	L	8.	L	H	L	L	(H)
					9.	L	L	H	H	(L)
					10.	H	L	L	L	(H)

Figure 5. Structure of item set used in Experiment 3.

In contrast, certain three-way interactions should be learned fairly easily within the context of a common-cause model. Figure 5 shows the structure of the material used in Experiment 3. The positive set was characterized by three correlated dimensions (H H H, or L L L), while the fourth dimension is irrelevant. This type of three-way interaction, like the pairwise interactions used in the correlated conditions in previous experiments, is consistent with a common-cause model in which the cause can vary in intensity. The negative set consisted of the full contrast set with respect to the first three dimensions, so that the subjects indeed had to learn the three-way interaction and could not use

two-way correlations to predict the correct response. In order to keep the negative set small, half of the subjects received the negative cases in which every uneven case has an L-value on the fourth irrelevant dimension, while for the other half these values were reversed. The dimensions and values were the same as those used in Experiment 1, with the addition of two levels of posture as the irrelevant dimension. As in the previous experiment, no hint was given regarding potential intensity variations of the virus. Two groups of subjects differed solely in the causal cover story they received: the disease story (common cause) or the emotional-response story (common effect). Twelve subjects served in each cover-story condition.

Results

The results presented in Figure 6 indicate that the three-way interaction was considerably more difficult to learn than were the correlated item sets within the earlier experiments with pairwise interactions (compare errors for the correlated conditions in Figures 3 and 4). Nonetheless, many subjects were able to attain the criterion of two passes through the items without an error, thus contradicting the implication of Gluck et al.'s (1989) configural-cue model, according to which the task should be unlearnable. In addition, and as in the previous experiments, the common-cause condition yielded a considerably lower error rate than did the common-effect condition, $F(1,22) = 5.40, p < .05$.

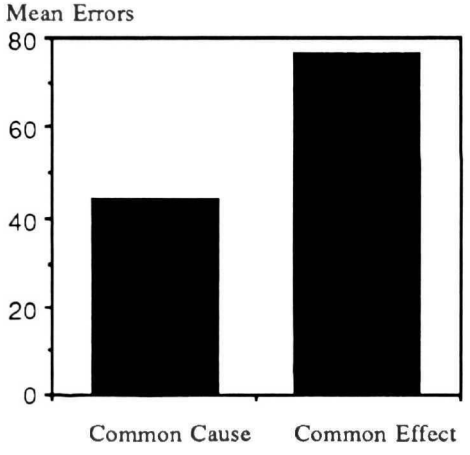


Figure 6. Mean errors prior to reaching criterion as a function of the causal model (common cause vs. common effect) for a correlated item set based on a three-way interaction in Experiment 3.

Discussion

Taken together, the three presented experiments clearly demonstrate the inadequacy of associationistic learning accounts of causal induction as they are embodied in recent connectionist models. Even though the cues and the required responses were identical across the two causal contexts, subjects proved sensitive to the structural implications of the different causal directions implied by the cover stories. Networks that simply code cues on the input layers and responses on the output layer cannot explain such reversal in the relative difficulty of linearly-separable versus correlated item sets, regardless of how they are internally configured.

Our results also demonstrate that causal induction cannot be reduced to associative learning. Associative accounts do not capture the fundamental differences between predictive and diagnostic reasoning (Pearl, 1988). Predictive reasoning requires learning the causal strengths between given causes and potential predicted effects. Once the causal links are learned, information about the presence of a cause allows probabilistic conclusions regarding its likely effects. In diagnostic reasoning, in which causes are inferred from effects, the situation is different. Even with perfect knowledge about cause-effect relationships, effect information is ambiguous with respect to its causes whenever there exists more than one potential cause. Reasoning in this situation requires an inference to the best explanation (Harman, 1986; Thagard, 1989). Different possible theories have to be weighed against each other, and a decision in favor of one or the other theory is based on the fit between the predictions of different theories and the evidence. An analogous approach is taken in research on statistical causal

models as they are embodied in linear structural equations (Bollen, 1989). We are currently modelling the differences between predictive and diagnostic reasoning within a symbolic-connectionist framework, exploring models in which units are interpreted as causes and effects and core links are viewed as causal connections.

Finally, our results are in agreement with many findings demonstrating an overall preference for linear models (e.g., Dawes, 1982; Trabasso & Bower, 1968). Learning linear models puts less strain on information processing because the impact of individual causes is not moderated by the presence of other causes. A number of philosophers have argued that common-cause structures are prevalent in scientific reasoning. Salmon (1984), in particular, argued that theoretical concepts play the role of common causes. Psychologists and philosophers have asked many times what we gain from inferring invisible entities. A possible answer, suggested by the present results, might be that inferred common causes help people to re-represent nonlinear observable structures within a basically linear mental model.

References

Bollen, K. A. (1989). *Structural equations with latent variables.* New York: Wiley.

Dawes, R. M. (1982). The robust beauty of improper linear models in decision making. In D. Kahneman, P. Slovic & A. Tversky (Eds.), *Judgment under uncertainty: Heuristics and biases* (pp. 391-407). Cambridge: Cambridge University Press.

Estes, W. K., Campbell, J. A., Hatsopoulos, N., & Hurwitz, J. B. (1989). Base-rate effects in category learning: A comparison of parallel network and memory storage-retrieval models. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15,* 556-571.

Gluck, M. A., & Bower, G. H. (1988a). Evaluating an adaptive network model of human learning. *Journal of Memory and Language, 27,* 166-195.

Gluck, M. A., & Bower, G. H. (1988b). From conditioning to category learning: An adaptive network model. *Journal of Experimental Psychology: General, 117,* 227-247.

Gluck, M. A., Hee, M. R., & Bower, G. H. (1989). A configural-cue network model of animal and human associative learning. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

Harman, G. (1986). *Change in view.* Cambridge, MA: MIT Press.

Minsky, M., & Papert, S. (1969). *Perceptrons: An introduction to computational geometry.* Cambridge, MA: MIT Press.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference.* San Mateo, CA: Morgan Kaufmann.

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and non-reinforcement. In A. H. Black & W. F. Prokasy (Eds.), *Classical conditioning II. Current research and theory.* New York: Appleton-Century-Crofts.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing* (Vol. 1, pp. 318-362). Cambridge, MA: MIT Press.

Salmon, W. C. (1984). *Scientific explanation and the causal structure of the world.* Princeton, NJ: Princeton University Press.

Shanks, D. R. (1990). Connectionism and human learning: Critique of Gluck and Bower (1988). *Journal of Experimental Psychology: General, 119,* 101-104.

Shanks, D. R., & Dickinson, A. (1987). Associative accounts of causality judgment. In G. Bower (Ed.), *The psychology of learning and motivation: Advances in research and theory, 21.* New York: Academic Press.

Shepard, R. N., Hovland, C. I., & Jenkins, H. M. (1961). Learning and memorization of classifications. *Psychological Monographs, 75,* 1-42.

Sutton, R. S., & Barto, A. G. (1981). Toward a modern theory of adaptive networks: Expectation and prediction. *Psychological Review, 88,* 135-170.

Thagard, P. (1989). Explanatory coherence. *Behavioral and Brain Sciences, 12,* 435-467.

Trabasso, T., & Bower, G. (1968). *Attention in learning: Theory and research.* New York: Wiley.

Wattenmaker, W. D., Dewey, G. I., Murphy, T. D., & Medin, D. L. (1986). Linear separability and concept learning: Context, relational properties, and concept naturalness. *Cognitive Psychology, 18,* 158-194.

Widrow, G., & Hoff, M. E. (1960). Adaptive switching circuits. *Institute of Radio Engineers, Western Electronic Show and Convention, Convention Record, 4,* 96-194.

Decision Models: A Theory of Volitional Explanation*

Ashwin Ram

Georgia Institute of Technology
School of Information and Computer Science
Atlanta, Georgia 30332-0280
(404) 853-9372
E-mail: ashwin@cs.gatech.edu

Abstract

This paper presents a theory of motivational analysis, the construction of volitional explanations to describe the planning behavior of agents. We discuss both the content of such explanations, as well as the process by which an understander builds the explanations. Explanations are constructed from *decision models*, which describe the planning process that an agent goes through when considering whether to perform an action. Decision models are represented as *explanation patterns*, which are standard patterns of causality based on previous experiences of the understander. We discuss the nature of explanation patterns, their use in representing decision models, and the process by which they are retrieved, used and evaluated.

1 Issues in explanation

In order to learn from experience, a reasoner must be able to *explain* what it does not understand. When a novel or poorly understood situation is processed, it is interpreted in terms of knowledge structures already in memory. As long as these structures provide expectations that allow the reasoner to function effectively in the new situation, there is no problem. However, if these expectations fail, the reasoner is faced with an *anomaly*. The world is different from its expectations. In order to learn from this experience, the reasoner needs to know *why* it made those predictions. It also needs to explain *why* the failure occurred, i.e., to identify the knowledge structures that gave rise to the faulty expectations, and to understand why its domain model was violated in this situation. Finally, it must store the new experience in memory for future use. Explanation is a central issue in this process of understanding and learning.

The construction of explanations is also known as *abduction*, or *inference to the best explanation*. This process is usually viewed as the chaining together of causal inference rules in order to create a causal chain, in which a proposed set of premises is shown to be causally responsible for the event or fact being explained. However, there are two problems with this view.

The first problem is the familiar one of combinatorial explosion of inferences. Most explanation programs create explanations by chaining together inference rules that describe the causality of the domain. For example, PAM [Wilensky, 1978] used a set of planning rules connecting together typical goals and plans of people, and chained them together to form motivational explanations for actions observed in a story. However, this process is very inefficient in complex domains, where the causal chains may be several steps long.

The second problem is the evaluation of explanations. Since the chaining process is seeking a connection between two concepts, most theories use an evaluation criterion based on the structural properties of this connection. For example, marker passing and spreading activation techniques (which are often proposed as a solution to the combinatorial explosion problem) usually judge the goodness of an explanation by the length of the causal chain. The shortest correct explanation is assumed to be the "best" one. However, the definition of "best" is dependent on the *goals* of the reasoner in forming the explanation and not just on the length or correctness of the causal chain underlying the explanation. In situations where there is no one "right" explanation, the "best" explanation must be more than a causal chain that describes the events in the domain; it must also address the reason that an explanation was required in the first place. This in turn determines what the reasoner can learn from the explanation.

*The research described was conducted while the author was at Yale University, and supported by the Defense Advanced Research Projects Agency and the Office of Naval Research under contract N00014-85-K-0108, and by the Air Force Office of Scientific Research under contracts F49620-88-C-0058 and AFOSR-85-0343.

In addition to processing issues, a theory of explanation must also address the content issues of the nature and representation of explanations. What is an explanation, and what kinds of knowledge does it provide? What is the nature of the causal knowledge that underlies volitional explanations? The answers to these questions depend both on the explanations that we desire to build, as well as on the process that is used to build them.

This paper presents a theory of explanation based on the claim that new explanations are built, not by chaining inference rules together, but rather by reusing explanations that have been encountered in previous situations and are already known to the system [Schank, 1986]. Our view raises several questions:

- *Content and representation:* What kinds of knowledge must an explanation provide? How do we represent this knowledge? What kinds of structures are used to represent explanations in memory? What is the vocabulary out of which these structures are built?
- *Retrieval:* How do we find pre-stored explanations in memory without having to try each one?
- *Evaluation:* How do we determine what kind of explanation is needed, and which explanation is the "best" one in a particular situation?
- *Learning:* How are explanations learned so that they can be reused in the future? What happens when pre-stored explanations don't apply to the current situation?[1]

The theory presented here has been implemented in the AQUA program, a story understanding program which learns about terrorism by reading newspaper stories about unusual terrorist incidents in the Middle East. We will illustrate our ideas with examples taken from this program. Further details may be found in [Ram, 1987; Schank and Ram, 1988; Ram, 1989].

2 What is an explanation?

The need for an explanation arises when some observed fact doesn't quite fit into the reasoner's world model, i.e., the reasoner detects an *anomaly*. An explanation is a knowledge structure that makes the anomaly go away. To illustrate the nature of such a structure, let us consider some candidate explanations for the following story (New York Times, Nov 27, 1985, page A9) from the domain of the AQUA program:

S-1: Suicide bomber strikes Israeli post in Lebanon.

SIDON, Lebanon, November 26 — A teenage girl exploded a car bomb at a joint post of Israeli troops and pro-Israeli militiamen in southern Lebanon today, killing herself and causing a number of casualties, Lebanese security sources said.

...

A statement by the pro-Syrian Arab Baath Part named the bomber as Hamida Mustafa al-Taher, born in Syria in 1968. The statement said she had detonated a car rigged with 660 points of explosives in a military base for 50 South Lebanon Army men and Israeli intelligence and their vehicles.

Why did Hamida go on the bombing mission?

(1) Because Lebanon is a Middle Eastern country.

(2) To destroy the Israeli military base.

(3) Because she was a religious fanatic.

(4) Because she didn't realize she was going to die during the mission.

Consider (1). This does not seem like an explanation for S-1. The reason isn't that (1) is false, but rather that there seems to be no causal connection between (1) and S-1. Thus it is not sufficient for a proposed explanation to be true; *an explanation must be causally connected to the anomaly.* It must contain a set of premises and a causal chain linking those premises to the anomalous proposition. If the reasoner believes the premises, the proposition ceases to be anomalous since the causal interactions underlying the situation can now be understood.

However, not all causal structures are explanations. For example, (2) is causally relevant to S-1, but it still doesn't feel like an explanation. To understand why, let us make the anomaly in S-1 explicit. The real question isn't "Why did Hamida go on the bombing mission?", but rather one of the following:

S-2: Why was Hamida willing to sacrifice her life in order to destroy the Israeli military base?

S-3: Why did Hamida go on a mission that would result in her own death?

The reason that explanation (2) feels strange is that it misses the point of the question. If the point is made explicit as in S-2, (3) is a possible explanation for the anomaly. Alternatively, if the real question is intended to be S-3, (4) is a possible explanation. The point is that, in order to qualify as an explanation, a causal description must address the underlying anomaly.

To state this another way, *an explanation must address the failure of the reasoner to model the situation correctly.* In addition to resolving the incorrect predictions, it must also point to the erroneous aspect of the chain of reasoning that led to the incorrect predictions. An explanation is *useful* if it allows the reasoner to learn and to improve its performance at its task; the claim here is that *an explanation must be both causal and relevant in order to be useful.* This is important in evaluating explanations to determine the best one for a particular situation.

[1] These issues are beyond the scope of this paper.

3 Explanation patterns

An explanation is a causal chain that demonstrates why the anomalous proposition might have occurred by introducing a set of premises that causally lead up to that proposition. There may be more than one explanation for a situation, depending on the question that the reasoner is interested in. For example, if the system needs to explain the motivations behind the girl's actions in story S-1, it may build what we think of as the *religious fanatic explanation*: The girl was a Moslem fanatic; she was so determined to further the cause of her religion that she was willing to die for it; and she believed that destroying the military base would help her religious cause.

The premise of this explanation is that the girl was a religious fanatic. If the reasoner believes or can verify the premises of an explanation, the conclusion is said to be explained. Explanations are often verbalized using their premises. Thus in normal conversation this explanation would be stated succinctly as "Because she was a religious fanatic." However, the real explanation includes the premises, the causal chain, and any intermediate assertions (such as the girl's belief that the bombing would help her religious cause) that are part of the causal chain.

How might a reasoner construct such an explanation? PAM [Wilensky, 1978] used a set of planning rules connecting typical goals and plans of people, and chained them together to form explanations such as the above. However, this is too inefficient in complicated situations, where the causal chains could be several steps long. To get around this problem, AQUA uses pre-stored explanations for stereotypical situations. These explanations represent standard patterns that are observed in these situations, and hence are called *explanation patterns* [Schank, 1986].

An explanation pattern (XP) is a stock explanation for a stereotypical situation. For example, *religious fanatic does terrorist act* is a standard XP many people have about the Middle East terrorism problem. One might think of them as the "scripts" of the explanation domain.[2] When a reasoner encounters a situation for which it has a canned XP, it tries to apply the XP to avoid detailed analysis of the situation from scratch.

This approach is known as *case-based explanation*, since previous cases or explanations known to the reasoner are used to help in the construction of new explanations. Explanatory cases in AQUA are based on the theory of explanation patterns described by [Schank, 1986], to which we add a theory of the representational structure and content of the XPs used in story understanding.

Explanations can be divided into two broad categories, physical and volitional.

3.1 Physical explanations

Physical explanations link events with the states that result from them, and further events that they enable, using causal chains similar to those of [Rieger, 1975] and [Schank and Abelson, 1977]. Physical explanations answer questions about the physical causality of the domain. For example, if the system had never read a story about a car bombing before, it might encounter an anomaly: "How can a car be used to blow up a building?" The answer to this question is a physical explanation:

(1) A car is a physical object.
(2) A car can contain explosives.
(3) A car can be propelled by driving it.
(4) Explosives can be blown up by the sudden impact of a car colliding with a building.
(5) A building can be blown up by blowing up explosives in its immediate vicinity.

Thus the explanation is that the bomber drove an explosive-laden car into the building, the impact caused the explosives to detonate, which caused the building to blow up.

3.2 Volitional explanations

Volitional explanations link actions that people perform to their goals and beliefs, yielding an understanding of the *motivations* of the characters. For example, the system might detect a different anomaly on reading story S-1, such as "Why would someone commit suicide if they are not depressed?" An explanation for this question, such as the religious fanatic explanation, must provide a motivational analysis of the reasons for committing suicide. For this reason, volitional explanations are also called motivational explanations. Although the basic structure of volitional explanations is the same as that of physical explanations, the vocabulary used to represent the causal chain is very different.

Volitional explanations fall into two broad categories:

1. **Abstract explanation patterns** for why people do things. These are standard high-level explanations for actions, such as "Actor does action because the outcome of action satisfies a goal of the actor."

2. **Stereotypical explanation patterns.** These are specific explanations for particular situation, such as "Shiite Moslem religious fanatic goes on suicide bombing mission."

[2] Unlike scripts, however, XPs are flexible since they contain a description of the *causality* underlying a situation in addition to a description of the situation itself. This allows XPs to be useful in novel situations, while retaining the advantages of pre-stored structures in stereotypical situations. The incremental elaboration of XPs in novel situations is discussed in [Ram, 1989; Ram, 1990b].

For example, an explanation of type 1 for the suicide bombing story could be "Because she wanted to destroy the Israeli base more than she wanted to stay alive." An explanation of type 2 would be simply "Because she was a religious fanatic." The internal causal structure of the latter explanation could then be elaborated to provide a detailed motivational analysis in terms of explanations of the first type if necessary.

Volitional explanations thus correspond to the filling out of the "belief-goal-plan-action" chain [Schank and Abelson, 1977; Wilks, 1977; Wilensky, 1978; Schank, 1986], although we need to expand the vocabulary of this chain in order to model such explanations adequately [Ram, 1989]. A volitional explanation relates the actions in which the characters in the story are involved to the *outcomes* that those actions had for them, the *goals*, *beliefs*, *emotional states* and *social states* of the characters as well as priorities or *orderings* among the goals, and the *decision process* that the characters go through in *considering* their goals, goal-orderings and likely outcomes of the actions before deciding whether to do those actions. A detailed volitional explanation involving the planning decisions of a character is called a *decision model*, and is illustrated in figure 1.

Decision models provide a theory of motivational coherence for stories involving volitional agents. When a decision model is applied to the actions of a given character in a story, it focusses attention on faulty assumptions or inconsistencies identified in the application of the decision model to the story. These inconsistencies signal anomalies, which must be explained by determining whether different parts of the decision model (e.g., the goals of the agent, his beliefs about the outcome, or his volition in deciding to perform the action) are actually present as assumed.

For example, the religious fanatic explanation is based on the following decision model:[3]

1. Explains: Why `volitional-agent` A did a `suicide-bombing` M, with `results` =
   (1) `death-state` of A
   (2) `destroyed-state` of `target`, a `physical-object` whose `owner` is an opponent religious group.
2. Premises:
   (1) A believes in the `religion` R.
   (2) A is a `religious-fanatic`, i.e., A has `high-religious-zeal`.
3. Internals:
   (1) A is religious and believes in the religion R (an `emotional-state`, perhaps caused by a `social-state`, such as upbringing).
   (2) A is strongly zealous about R (an `emotional-state`).
   (3) A wants to spread his religion R (a `goal`, initiated by (1) and (2)).
   (4) A places a high priority on his goal in (3), and is willing to sacrifice other goals which we would normally place above the religion goal (a `goal-ordering`, initiated by (1) and (2)).
   (5) A believes that performing a suicide bombing against opponent religious groups will help him achieve his goal in (3) (a belief or `expected-outcome`).
   (6) A knows that the performance of a suicide bombing may result in a negative outcome for him (an `expected-outcome`).
   (7) A weighs his goals (3), goal-orderings (4), and likely outcomes (5) and (6) (a `consideration`).
   (8) A decides to do the suicide bombing M (a `decision`, based on the considerations in (7)).
   (9) A does the suicide bombing M (an action or `mop`, whose `actor` is A).
   (10) The suicide bombing has some outcome for A, which is either positive or negative as viewed from the point of view of A's goals and goal-orderings (a `self-outcome`).

The representation of the religious fanatic explanation is shown in figure 2. The decision model has the following components:

The outcome of an action: Every action results in some set of states that may or may not be beneficial to the people involved in that action, depending on their goals at that time. The outcome of an action, therefore, must be modelled *from the point of view of a particular volitional agent* involved in that action. The most common volitional participants are `actor` and `planner`, but any role involving a volitional agent must potentially be explained.

The decision process: Every agent involved in an action makes a *decision* about whether to participate in that particular volitional role (`actor`, `planner`, `object`, etc.) in the action. Such decisions represent the *planning* process that the agent underwent prior to the action. A complete model of this process requires a sophisticated vocabulary of goals, goal interactions, and plans, such as that of [Wilensky, 1983] or [Hammond, 1986]. There are three basic kinds of decisions:

1. **Choice:** The agent *chooses* to participate or not to participate in a given volitional role in some action. The explanation must describe why he made this choice.
2. **Agency:** The agent is *induced* to participate or not to participate in a given volitional role in an action. This is similar to the previous case in that the agent "enters" the action of his own volition. The difference is that here the agent is acting under the agency of another agent. Thus the

[3] `Typewriter font` represents actual vocabulary items used by the AQUA program. Further details of the representation may be found in [Ram, 1989].

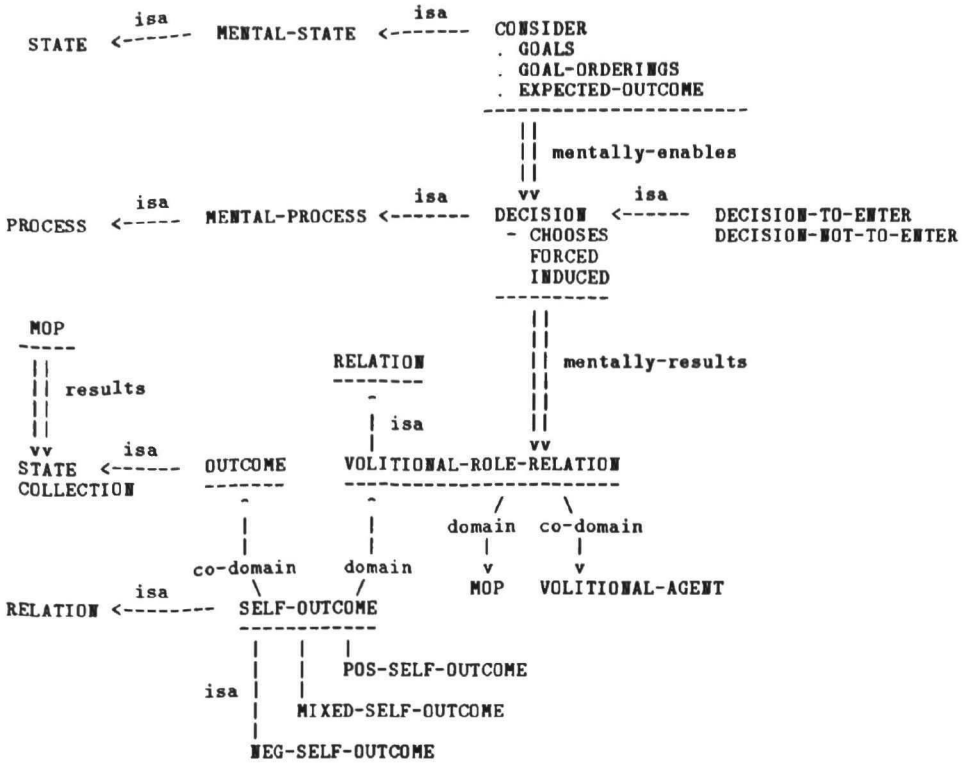


Figure 1: The structure of volitional explanations. A **volitional-agent** participates in some **volitional-role** in a **mop**, which then **results** in an **outcome** (a **collection** of **states**). Prior to this, the **volitional-agent** undergoes a **decision** process in which he **considers** his **goals**, **goal-orderings** and **expected-outcome**, which then **mentally-results** in the **volitional-role-relation** being considered being true (**in**) or false (**out**) depending on the outcome of the **decision**.

explainer must be able to model inter-agent interactions [Schank and Abelson, 1977; Wilensky, 1983; Ram, 1984].

3. **Coercion:** The agent is *forced* to participate or not to participate in a given volitional role in an action. This case arises when an agent is physically coerced into participation or non-participation.

Considerations in decisions: The system also needs to reason about what an agent was considering as he made a particular decision. Considerations model the goals and beliefs of an agent, along with orderings among these goals and expected outcome of the action being considered. Considerations are composed of three constituents: (1) **goals** considered by the agent while deciding whether or not to participate in an action, (2) **goal-orderings**, the agent's prioritization of these goals, and (3) the **expected-outcome**: the agent's beliefs about what the outcome of the action is likely to be. This is represented by the **consider** node in figure 1.

Each of these constituents may itself need to be explained further. For example, the system might question the social or mental (e.g., emotional) states that **initiated** a particular goal or goal-ordering in an agent, or how a particular belief about the outcome of an action came about. Explanations, therefore, may need to be *elaborated* according to the demands of the story and the goals of the system.

4 Structure of explanation patterns

AQUA has several XPs indexed in memory, representing its causal knowledge of the terrorism domain. These XPs are represented as graph structures (as illustrated above) with four main components:

1. **PRE-XP-NODES:** Nodes that represent what is known before the XP is applied. One of these nodes, the EXPLAINS node, represents the particular action being explained.
2. **XP-ASSERTED-NODES:** Nodes asserted by the XP as the explanation for the EXPLAINS node. These comprise the premises of the explanation.
3. **INTERNAL-XP-NODES:** Internal nodes as-

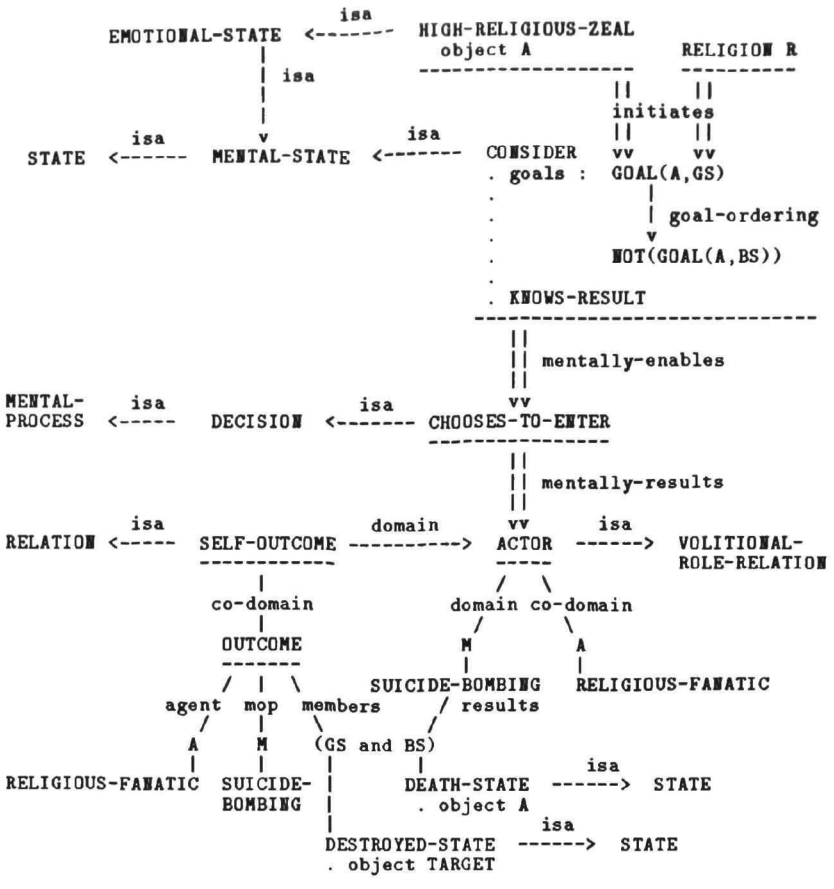


Figure 2: The religious fanatic explanation pattern. A is the agent, R his religion, M the action he chooses to do, and GS and BS the good and bad outcomes for A as a result of doing that action. A volitionally chooses to perform M knowing both outcomes, the death-state of A and the destroyed-state of the target.

serted by the XP in order to link the XP-ASSERTED-NODES to the EXPLAINS node.

4. **LINKS:** Causal links asserted by the XP. These taken together with the INTERNAL-XP-NODES are also called the internals of the XP.

An explanation pattern states that the XP-ASSERTED-NODES lead to the EXPLAINS node (which is part of a particular configuration of PRE-XP-NODES) via a set of INTERNAL-XP-NODES, the nodes being causally linked together via the LINKS (which in turn could invoke further XPs). In other words, an XP represents a causal chain composed of a set of nodes connected together using a set of LINKS (causal rules or XPs). The "antecedent" (or premise) of this causal chain is the set of XP-ASSERTED-NODES, the "internal nodes" of the causal chain are the INTERNAL-XP-NODES of the XP, and the "consequent" is the EXPLAINS node. The difference between XP-ASSERTED-NODES and INTERNAL-XP-NODES is that the former are merely asserted by the XP without further explanation, whereas the latter have causal antecedents within the XP itself.

5 The explanation cycle

An explanation-based understander must be able to detect anomalies in the input, and resolve them by building motivational and causal explanations for the events in the story in order to understand why the characters acted as they did, or why certain events occurred or did not occur. This process characterizes both "story understanders" that try to achieve a deep understanding of the stories that they read, as well as programs that need to understand their domains in service of other problem-solving tasks. Explanations are constructed by retrieving XPs from memory, applying them to the situation at hand, and verifying or evaluating the resulting hypotheses.

5.1 Anomaly detection

Anomaly detection refers to the process of identifying an unusual fact that needs explanation. The

anomalous fact may be unusual in the sense that it violates or contradicts some piece of information in memory. Alternatively, the fact may be unusual because, while there is no explicit contradiction, the reasoner fails to integrate the fact satisfactorily in its memory.

5.2 Explanation pattern retrieval

When faced with an anomalous situation, the reasoner tries to retrieve one or more explanation patterns that would explain the situation. Ideally, an XP should be indexed in memory such that it is retrieved only in those situations in which it is applicable. But this is impossible in practice. For example, consider the applicability conditions for "blackmail." In general, blackmail is a possible explanation whenever "someone does something he doesn't want to do because not doing it results in something worse for him." But trying to show this in general is very hard. Thus, in addition to general applicability conditions, a reasoner must learn specific, sometimes superficial, features that suggest possibly relevant XPs even though they may not completely determine the applicability of the XP to the situation. For example, a classic blackmail situation is one where a rich businessman who is cheating on his wife is blackmailed for money using the threat of exposure. If one read about a rich businessman who suddenly began to withdraw large sums of money from his bank account, one would expect to think of the possibility of blackmail. However, one does not normally think of blackmail when one reads a story about suicide bombing, although theoretically it is a possible explanation.

AQUA indexes motivational XPs in memory using typical contexts in which the XPs might be encountered (*situation indices*), as well as character stereotypes representing typical categories of people to whom the XPs might be applicable (*stereotype indices*) [Ram, 1989]. The third type of index is known as the *anomaly index* or *category index*. Recall that in addition to explaining the occurrence of the event, it is important for the XP to address the anomaly which arose from the failure of the reasoner to model the situation correctly. Thus the type of the anomaly provides an index to the type of XP required to build an explanation. For example, if the anomaly was one where an actor performed an action that violated one of the actor's own goals, the reasoner might look for a "goal sacrifice" XP (such as a religious fanatic sacrificing her life for the cause of her religion), or an "actor didn't know outcome" XP (such as a gullible teenager not realizing what the outcome of her action was going to be). However, the category of goal sacrifice XPs would be inappropriate for an anomaly in which the actor failed to perform an action which only had a good outcome for the actor; in this case, a "missed opportunity" XP might be chosen.

5.3 Explanation pattern application

Once a set of potentially applicable XPs is retrieved, the reasoner tries to use them to resolve the anomaly. This involves instantiating the XPs, filling in the details through elaboration and specification, and checking the validity of the final explanations. An XP is instantiated by unifying the EXPLAINS node of the XP with the description of the situation being explained, and instantiating the INTERNAL-XP-NODES and LINKS. If all the PRE-XP-NODES and INTERNAL-XP-NODES of the XP fit the situation, the hypothesis is applicable. If the unification fails, the hypothesis is rejected.[4]

5.4 Hypothesis verification and evaluation

The final step in the explanation process is the confirmation or refutation of possible explanations, or, if there is more than one hypothesis, discrimination between the alternatives. A hypothesis is a causal graph that connects the premises of the explanation to the conclusions via a set of intermediate assertions. At the end of this step, the reasoner is left with one or more alternative hypotheses. Partially confirmed hypotheses are maintained in a data dependency network called a *hypothesis tree*, along with questions (unconfirmed XP-ASSERTED-NODES) representing what is required to verify these hypotheses.

There are five criteria for evaluating the goodness of an explanation:

1. **Believability:** Does the system believe the XP from which the hypothesis was derived? This is not an issue when all XPs in memory are believed, but for a program that learns new XPs, some of which may be incomplete, the believability of the XP is an important criterion in deciding whether to believe the resulting hypothesis.
2. **Applicability:** How well does the XP apply to this situation? Did it fit the situation without any modifications?
3. **Relevance:** Does the XP address the underlying anomaly? Does it address the knowledge goals of the reasoner (i.e., does it allow the reasoner to learn)?
4. **Verification:** How definitely was the explanation confirmed or refuted?
5. **Specificity:** How specific is the XP? Is it abstract and very general (e.g., a proverb), or is it detailed and specific?

Intuitively, a "good" explanation is not necessarily one that can be proven to be "true" (criterion

[4] There is also the possibility of modifying the hypothesis to fit the situation [Schank, 1986; Kass *et al.*, 1986].

4), but also one that seems plausible (**1** and **2**), fits the situation well (**2** and **5**), and is relevant to the goals of the reasoner (criterion **3**).

The relevance criterion is important if the explanation is created for some purpose (and not as an end in itself). The fact that the reasoner encountered an anomaly indicates a need to learn, which could arise in several ways. The reasoner may not have the knowledge structures to deal with a novel situation, or the knowledge structures that the reasoner applies to the situation may be incomplete or incorrect. The domain knowledge may be mis-indexed in memory, i.e., the reasoner may have the knowledge structures to deal with the situation, but it may be unable to retrieve them since they are not indexed under the cues that the situation provides.

When an explanation is built, the reasoner needs to be able to identify the kind of processing error that occurred and invoke the appropriate learning strategy. For example, if an incomplete knowledge structure is applied to a situation, the resulting processing error represents both the knowledge that is missing, as well as the fact that this piece of knowledge, when it comes in, should be used to fill in the gap in the original knowledge structure. Similarly, if an error arose due to a mis-indexed knowledge structure, the explanation, when available, should be used to re-index the knowledge structure appropriately. The explanation is therefore constrained by the needs of the learning process [Ram, 1990a].

6 Conclusion

Abduction, or inference to the best explanation, is a central component of the reasoning process. Abduction is viewed, not as a process of chaining together inference rules to produce causal chains, but rather one of case-based reasoning from pre-stored causal chains, known as explanation patterns, associated with prior experiences in memory. This provides a way to control the combinatorial explosion of inferences, but introduces a new set of issues: the content and representation of explanation patterns, the types of indices used to retrieve XPs from memory, the evaluation of candidate hypotheses, and the learning of new XPs.

Evaluation is facilitated by using anomaly characterizations as retrieval indices for XPs. The "best" explanation is not one that is the most "correct," if correctness is even measurable in the domain of interest, but one that is most useful to the process that is seeking the explanation. The anomaly detection process provides retrieval cues that are used to find explanation patterns that are likely to be relevant to the anomaly.

These ideas have been explored in the AQUA program, a computer model of the theory of question-driven understanding. AQUA learns about terrorism by reading newspaper stories about terrorist incidents in the Middle East. The requirements of this task provided constraints on the theory of explanation presented here.

References

[Hammond, 1986] K. J. Hammond. *Case-Based Planning: An Integrated Theory of Planning, Learning and Memory.* Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, October 1986. Research Report #488.

[Kass *et al.*, 1986] A. Kass, D. Leake, and C. Owens. *SWALE: A Program That Explains*, pages 232–254. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[Ram, 1984] A. Ram. Modelling Characters and their Decisions: A Theory of Compliance Decisions. Master's thesis, University of Illinois at Urbana-Champaign, Urbana, IL, August 1984. Technical Report T-145.

[Ram, 1987] A. Ram. AQUA: Asking Questions and Understanding Answers. In *Proceedings of the Sixth Annual National Conference on Artificial Intelligence*, pages 312–316, Seattle, WA, July 1987. American Association for Artificial Intelligence, Morgan Kaufman Publishers, Inc.

[Ram, 1989] A. Ram. *Question-driven understanding: An integrated theory of story understanding, memory and learning.* Ph.D. thesis, Yale University, New Haven, CT, May 1989. Research Report #710.

[Ram, 1990a] A. Ram. Goal-Based Explanation. In *Proceedings of the AAAI Spring Symposium on Automated Abduction*, Palo Alto, CA, March 1990.

[Ram, 1990b] A. Ram. Incremental Learning of Explanation Patterns and their Indices. In *Proceedings of the Seventh International Conference on Machine Learning*, Austin, TX, June 1990.

[Rieger, 1975] C. Rieger. Conceptual Memory and Inference. In R. C. Schank, editor, *Conceptual Information Processing.* North-Holland, Amsterdam, 1975.

[Schank and Abelson, 1977] R. C. Schank and R. Abelson. *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures.* Lawrence Erlbaum Associates, Hillsdale, NJ, 1977.

[Schank and Ram, 1988] R. C. Schank and A. Ram. Question-driven Parsing: A New Approach to Natural Language Understanding. *Journal of Japanese Society for Artificial Intelligence*, 3(3):260–270, May 1988.

[Schank, 1986] R. C. Schank. *Explanation Patterns: Understanding Mechanically and Creatively.* Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[Wilensky, 1978] R. Wilensky. *Understanding Goal-Based Stories.* Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1978.

[Wilensky, 1983] R. Wilensky. *Planning and Understanding.* Addison-Wesley, Reading, MA, 1983.

[Wilks, 1977] Y. Wilks. What Sort of Taxonomy of Causation Do We Need for Language Understanding. *Cognitive Science*, 1:235, 1977.

Knowledge Goals: A Theory of Interestingness*

Ashwin Ram

Georgia Institute of Technology
School of Information and Computer Science
Atlanta, Georgia 30332-0280
(404) 853-9372
E-mail: ashwin@ics.gatech.edu

Abstract

Combinatorial explosion of inferences has always been one of the classic problems in AI. Resources are limited, and inferences potentially infinite; a reasoner needs to be able to determine which inferences are useful to draw from a given piece of text. But unless one considers the goals of the reasoner, it is very difficult to give a principled definition of what it means for an inference to be "useful."

This paper presents a theory of inference control based on the notion of interestingness. We introduce *knowledge goals*, the goals of a reasoner to acquire some piece of knowledge required for a reasoning task, as the focussing criteria for inference control. We argue that knowledge goals correspond to the *interests* of the reasoner, and present a theory of interestingness that is functionally motivated by consideration of the needs of the reasoner. Although we use story understanding as the reasoning task, many of the arguments carry over to other cognitive tasks as well.

1 Cognitive motivations: Knowledge goals as a basis for interestingness

When we compare the way people read newspaper stories with how computer programs typically read them, we notice the following differences:

Subjectivity: People are biased. They interpret stories in a manner that suits them. They jump to conclusions. Computer programs, on the other hand, are usually designed to read stories in an objective manner, and to extract the "correct" or "true" interpretation of a story to the extent that they can.

Variable depth parsing: People don't read everything in great detail. They concentrate on details that they find relevant or interesting, and skim over the rest. In contrast, computer programs are designed to attend to every aspect of a story that is within the scope of their knowledge structures. Consequently, they either process the entire story in great depth, or else they skim everything in the story. They can not decide which aspects to process in detail and which ones to ignore.

Learning and change: People change as they read. They never read the same story twice in the same way. They notice different things the second time around, or they simply get bored. After reading a story, they interpret other similar stories differently. Computer programs, in contrast, are not adaptive; they always read a given story the same way.

What makes people different from computer programs? What is the missing element that our theories don't yet account for? The answer is simple: *People read newspaper stories for a reason: to learn more about what they are interested in.* Computers, on the other hand, don't. In fact, computers don't even have interests; there is nothing in particular that they are trying to find out when they read. If a computer program is to be a model of story understanding, it should also read for a "purpose."

Of course, people have several goals that do not make sense to attribute to computers. One might read a restaurant guide in order to satisfy hunger or entertainment goals, or to find a good place to go for a business lunch. Computers do not get hungry, and computers do not have business lunches.

However, these *physiological* and *social goals* give rise to several *intellectual* or *cognitive goals*. A goal to satisfy hunger gives rise to goals to find information: the name of a restaurant which serves the desired type of food, how expensive the restaurant is, the location of the restaurant, etc. These are goals to acquire information or knowledge, and are called *knowledge goals*. These goals can be held by computers too; a computer might "want" to find out the location of a restaurant, and read a guide in order to do so in the same way as a person might. While such a goal would not arise out of hunger in the case of the computer, it might well arise out of the "goal" to learn more about restaurants.

In other words, knowledge goals also arise from

*The research described was conducted while the author was at Yale University, and supported by the Defense Advanced Research Projects Agency and the Office of Naval Research under contract N00014-85-K-0108, and by the Air Force Office of Scientific Research under contracts F49620-88-C-0058 and AFOSR-85-0343.

the desire to learn, to pursue one's intellectual interests, to improve one's model of the world. These goals can be viewed as *questions* about the domain of interest. To be interested in terrorism, for example, is to have a lot of questions about various aspects of terrorism, and to think about these questions in the context of input data about terrorism, such as newspaper stories about terrorist incidents. The point of reading these stories is to answer one's questions, as well as to reveal flaws or gaps in one's model of terrorism in order to try to improve this model. These gaps give rise to new questions which in turn stimulate further interest in terrorism. Both computers and people can be "interested" in terrorism in this sense.

In contrast with people, therefore, a computer has only one underlying goal: to learn and improve its world model.[1] However, this (and, in the case of people, other physical and social goals) gives rise to knowledge goals that then drive the understanding process.

2 Computational motivations: What are the knowledge goals of an understanding program?

Understanding, then, can be viewed as the pursuit of one's interests or questions. However, it would defeat the purpose to build a "question-asking" or "interest-pursuing" program per se. Instead, these questions and interests should arise naturally as *cognitive goals* of the program during various stages of the reasoning process. This means that the program should ask a question only when it has a need to acquire that piece of knowledge. For example, in the case of a story undertanding system, a knowledge goal should be formulated only when the system needs to know the answer for the purposes of understanding the story. In other words, knowledge goals should be *functionally useful* to the overall goals of the system.

The theory of knowledge goals presented in this paper depends on a theory of *understanding tasks*, the basic tasks of an understander. In addition to parser-level tasks such as noun group connection, pronoun reference, etc., these tasks include the integration of facts with what the understander already knows, the detection of anomalies in the text which identify flaws or gaps in the understander's model of the domain, the formulation of explanations to resolve those anomalies, the confirmation and refutation of potential explanations, the learning of new explanations for use in understanding future situations. These are the basic tasks that an understander needs to be able to perform.

In order to carry out these tasks, the understander needs to integrate the text, which is often ambiguous, elliptic and vague, with its world knowledge, which is often incomplete. In formulating an explanation, for example, the understander may need to know more about the situation than is explicitly stated before it can decide which is the best explanation. However, it is impossible to anticipate when a particular piece of knowledge will be available to the understander, since the real world (in the case of a story understanding program, the story) will not always provide exactly that piece of knowledge at exactly the time that the understander requires it. *Thus the understander must be able to suspend questions in memory, and reactivate them at the right time when the information it needs becomes available.* In other words, the understander must be able to remember what it needs to know, and why.

Furthermore, the system's understanding of any real world domain can never be quite complete. Conventional script, frame or schema-based theories assume that understanding means finding an appropriate script, frame or schema in memory and fitting it to the story. Schemas in memory are assumed to be "correct;" if an applicable schema is found, the story is understood. However, this model is inadequate since an understander's memory is always incomplete. Knowledge structures often have gaps in them, especially in poorly understood domains. These gaps correspond to what the understander has not yet understood about the domain. Even if a schema appears to be correct, novel experiences or stories may reveal flaws in the schema or a mismatch with the real world. Furthermore, the schema may not be indexed correctly in memory.

Understanding tasks, therefore, generate information subgoals or questions, representing what the understander needs to know in order to carry out the current task, be it explanation, learning, or any other cognitive task. These questions constitute the specific knowledge goals of the system, and are used to focus the understanding process.

Our theory of knowledge goals is motivated by these functional considerations, and corresponds well with a theory of interestingness motivated by the above cognitive considerations. The theory has been implemented in a computer program called AQUA (Asking Questions and Understanding Answers), which learns about terrorism by reading newspaper stories about unusual terrorist incidents in the Middle East [Ram, 1987; Schank and Ram, 1988; Ram, 1989]. AQUA uses its knowledge goals to direct the understanding process. We will illustrate our ideas with examples taken from this program.

3 A taxonomy of knowledge goals

Knowledge goals can be characterized according to the type of understanding task that they arise from.

Text goals: Knowledge goals of a text analysis program, arising from text-level tasks. These are the questions that arise from basic syntactic and semantic analysis that needs to be done on the input

[1] Since computers will eventually be expected to interact with the physical world (e.g., robots) and the social world (e.g., employees), they will also be expected to have some of the physical or social goals that we currently attribute only to people.

text, such as noun group attachment or pronoun reference.

Memory goals: Knowledge goals of a dynamic memory program, arising from memory-level tasks. A dynamic memory must be able to notice similarities, match incoming concepts to stereotypes in memory, form generalizations, and so on.

Explanation goals: Goals of an explainer that arise from explanation-level tasks, including the detection and resolution of anomalies, and the building of motivational and causal explanations for the events in the story in order to understand why the characters acted as they did, or why certain events occurred or did not occur.

Relevance goals: Goals of any intelligent system in the real world, concerning the identification of aspects of the current situation that are "interesting" or relevant to its own goals.

AQUA is an implementation of a integrated theory of story understanding, memory and learning, called *question-driven understanding*, which addresses the above issues. In addition to their theoretical role in our model of inference control and interestingness, knowledge goals have also played an implementational role in our research by providing a uniform mechanism for the integration of various cognitive processes. For example, knowledge goals arising from, say, memory tasks are indexed in memory and used in the same way as knowledge goals arising from explanation tasks. A knowledge goal generated from one task may be suspended, and satisfied opportunistically during the pursuit of some other task at a later stage or even during the processing of a different story. Implementational details may be found in [Ram, 1989].

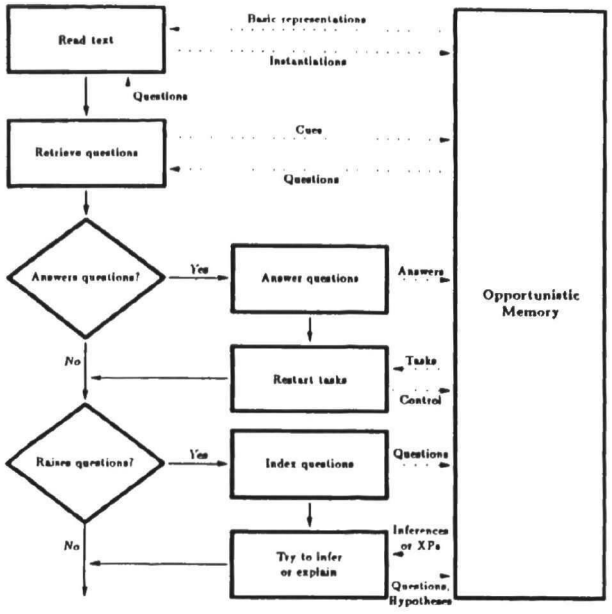


Figure 1: Control structure: The understanding cycle. A fact is interesting if it satisfies a knowledge goal pending in memory, or if it gives rise to new knowledge goals. Uninteresting facts pass vertically down with minimal processing; interesting facts cause suspended understanding tasks to be restarted, or new tasks to be created. New tasks can give rise to new knowledge goals, which are suspended along with the tasks if answers are not yet known and cannot be inferred.

4 Using knowledge goals to guide processing

A program that uses knowledge goals to guide understanding is an improvement over one that processes everything in equal detail, i.e., one that is completely text-driven. An understander that is completely text-driven would process everything in detail in the hope that it might turn out to be relevant. To avoid this, the understander should draw only those inferences which would help it find out what it needs to know. In other words, the understander should use its knowledge goals to focus its attention on the interesting aspects of the story, where "interesting" can be defined as "relating to something the understander wants to find out about."

Why would an understander need to find something out in the first place? Ultimately, the point of reading is to learn more about the world. Questions arise when reading a story reveals gaps or inconsistencies in the world model. It is useful to focus attention on such questions because they arise from a "need to learn." For example, questions arising from anomalous facts are more useful than those arising from routine stereotypical facts, since in the former case the understander may learn something new about the world.

There are two basic ways in which a fact can turn out to be worth processing:

Top-down: A fact that answers a question is worth focussing on since it helps to achieve a knowledge goal of the understander, which in turn allows the understander to continue the reasoning task that was awaiting the answer.

Bottom-up: A fact that raises new questions is worth focussing if the questions arise from a gap or inconsistency in the understander's knowledge base, since the understander may be able to improve its knowledge base by learning something new about the world.

These correspond to the two diamonds in figure 1. These diamonds attempt to determine which facts the understander should focus on. To improve on this even further, the understander needs a way of determining which knowledge goals worth pursuing and which ones are not. Not all questions are equally important, nor are all answers equally valuable. The understander needs to be able to determine the priorities of its knowledge goals, depending

on how likely the understander is to learn something by thinking about these knowledge goals. These decisions are made using a set of heuristics that will be described below.

The decision to focus attention corresponds closely with the notion of "interestingness." When an understander focuses on a particular fact and processes it in greater detail, it can be said to be "interested" in that fact.[2] For this reason, focus of attention heuristics can also be thought of as *interestingness heuristics*. These heuristics provide a functional definition of "interestingness" as a criterion for focussing attention: *Interestingness is a guess at what one thinks one might learn from paying attention to a fact or a question.* The guess must be made without processing the fact or question in detail, because otherwise the purpose of focussing attention to control inferences would be defeated. Thus the interestingness heuristics described below are indeed *heuristics* rather than precise measures of the value of thinking about a fact or a question.

5 Interestingness heuristics

In order to use questions to control inferences, an understander must be able to determine the interestingness of questions based on their content, as well as the task that they arose from. It must also be able to identify facts in the story that are interesting by virtue of being relevant to questions that the understander is interested in.

There are two types of heuristics for determining interestingness:[3]

Content-based: The interestingness of some input depends on its content or domain (more specifically, on the relationship between its content and the system's goals). In other words, some things are more interesting than others, depending on their relationship to the system's goals. For example, if one is intending to fly KLM in the near future, a story about a KLM flight being hijacked would be very interesting even if it were a stereotypical hijacking story. The understander would try to draw those inferences that were relevant to its goal of flying KLM. Similarly, stories about people one personally knows are more interesting than stories about strangers.

These heuristics use the content of the fact or question to determine its interestingness. The issue here is, which particular facts should the understander focus on? Which particular facts does the understander need to learn about?

Structure- or Configuration-based: Some kinds of situations are more interesting than others. For example, expectation failures are interesting, regardless of the content of the particular expectation that failed. These heuristics use the structure of the knowledge to determine interestingness. The issue here is, can the structure of the situation be used to determine which aspects of the situation are worth focussing on? Which configurations of knowledge structures signal gaps that the understander needs to learn about?

Of course, in any particular situation, the two conditions need to be combined in order to determine the overall interestingness of the input. For example, an expectation failure that relates to a goal of the system would have a higher priority than one that does not, and so the system would be more interested in the former.

Both types of heuristics identify situations in which the understander might be able to learn something useful. In the first case, the understander might learn more about a person it knows, or it might learn a new way to achieve a goal that it has. In the second case, the understander might be able to update its world model by identifying gaps in its model.

An example of the first type of heuristic is the principle of goal identification used by the POLITICS program [Carbonell, 1979]:

> If the understander of an event identifies with the goals of one of the actors, he will focus attention on inferences that lead to the fulfillment of these goals.

This is a content-based heuristic since it relies on the actual type of goal, not merely on the fact that there is a goal being pursued. A configuration-based heuristic, on the other hand, relies on particular relationships between concepts in memory, not specifically on what those concepts are. For example, POLITICS used the following configuration-based heuristic to focus its attention:

> Objective/Means distinction: If there are two or more actions in an event, and some actions are instrumental to stated or implicit objectives of one of the actors, the understander should focus attention on the objectives and non-instrumental actions.

In other words, an instrumental action is less likely to be significant than a larger action that it is part of, regardless of what the particular actions are. The reason this is a good strategy for an understander, according to the definition of interestingness proposed earlier, is that the understander is more likely to learn something by thinking about the larger action than it is by thinking about the instrumental action.

AQUA uses several heuristics to judge interestingness. These heuristics can be categorized according to the type of understanding goals that they pertain to. Let us start with relevance goals.

[2]Since interestingness depends on one's goals, the heuristics presented here do not cover interests that arise from goals that lie outside the scope of the basic understanding and learning tasks that AQUA performs. For example, a parent would be interested in the report card of his child. Since AQUA's goals do not include caring for children, it would not have any reason to be interested in a report card, unless the report card was anomalous with respect to AQUA's beliefs.

[3]This is orthogonal to the top-down/bottom-up distinction made earlier.

5.1 Interestingness from relevance goals

Interestingness arising from personal relevance usually falls into the class of content-based interestingness, since particular goals, people, locations, etc. are identified as being interesting to the understander. Questions and facts involving these goals, people, locations, etc. are worth pursuing since the understander might learn something relevant to it by doing so.

5.1.1 What could be relevant to a program? In order for something to be personally relevant to a program, the program must have a personality in the real world. There must be goals it wants to achieve, people it knows or has heard of, places it knows about or has grown up in, and so on. Stories relevant to this personality are interesting even if they do not involve anomalies or novel explanations.

For example, two of the focus of attention criteria used by POLITICS fall into this category: "goal identification" and "interest in VIP activities." Since POLITICS had a political ideology, it could be said to have a personality in the sense used here. Thus it could focus its attention on those aspects of a situation that were relevant to its goals.

Since AQUA does not have any real experiences outside of reading stories, its "personality" consists of its knowledge goals, i.e., the questions that it is interested in finding answers to. The people, institutions, objects and locations that it is interested in learning about are the people, institutions, objects and locations that are involved in these questions. The goals that are interesting to AQUA are goals of characters in these stories that it has questions about.

One could say that AQUA has "adopted" the goals that it has questions about, in the sense that it is interested in stories (or aspects of stories) about such goals as if they were its own (an example follows later). An alternative approach to the problem of where a program might get its goals is to "give" goals to the program, for example, by programming particular ideologies into the program as in POLITICS. In AQUA, this would be analogous to tagging particular goals or people as being "personally relevant" or "interesting" to the program.

In contrast, the approach used in AQUA is to allow the program to evolve its own set of interests that are functional to the purpose of the program (learning about terrorism), by letting the questions that arise from this purpose *be* its goals. These interests can be used to focus attention on those aspects of the story that would help it achieve its purpose.

Let us now discuss interestingness heuristics based on this notion of personal relevance.

5.1.2 Goal relevance. This is similar to the "goal identification" criterion used by POLITICS.

H-1: Goal relevance
A fact that could be instrumental to or could hinder a goal of the understander is more interesting than one that has no relevance to the understander's goals. A fact that directly matches or conflicts with a goal of the understander is very interesting.

5.1.3
Vicarious goals. Vicarious interests arise from goal configurations that are similar to one's own, or to those one is likely to have at some point. Stories about crisis goals or sudden goal changes, for example, are usually interesting from a vicarious point of view since the understander is likely to experience similar crisis goals or goal changes.

These heuristics are configuration-based, as opposed to content-based, since all goal changes are inherently interesting regardless of the particular goals involved. However, goal changes involving goals of personal relevance to the understander would be more interesting than those involving other goals.

H-2: Vicarious goal change
An action that drastically changes the goals of the planner or actor of the action is interesting.

H-3: Vicarious crisis goal
An action that initiates a crisis goal of the planner or actor of the action is interesting.

An example of a vicarious crisis goal is the self-preservation goal arising from acts of violence. Schank calls violence an "absolute interest" [Schank, 1979] since people universally seem to be interested in violence. This follows from the application of heuristic H-3 to the thematic goal of self-preservation which is universal among people.

5.1.4 Actor relevance. There are four degrees to which a particular person or institution can be relevant to an understander, each of which produces more interest than the previous one:

H-4: Actor relevance
Stories involving people that are completely unknown are the least interesting. More interesting than these are stories involving people who the understander has heard of. Still more interesting are stories involving people who are famous, such as celebrities. Stories involving people who the understander personally knows are the most interesting.

This heuristic is similar to the "actor relevance" and "interest in VIP activities" criteria of POLITICS. AQUA uses similar heuristics for object and location relevance, omitted here due to space limitations.

5.2 Interestingness from explanation goals

So far, we have seen interestingness heuristics that tried to identify situations involving goals, people, objects or places that are personally relevant to the understander. Such situations provide an opportunity to learn something of personal relevance, e.g.,

to learn something more about a person the understander is interested in, or to learn a new plan to achieve a goal that the understander has. There is another class of heuristics which identify potential gaps in the understander's knowledge, in order to determine what the understander might learn from processing the given situation. Situations that are interesting according to these heuristics are those that allow the understander to improve its world model. These heuristics are based on explanation and memory goals of the understander.[4] Let us start with explanation goals.

5.2.1 Anomaly detection. Anomalies arise when incoming facts do not fit in with what the understander expected to see. Anomalies are interesting because there is a possibility that the world model that underlies the failed expectations is incorrect, which signals a need to learn more about the domain.

H-5: Anomaly detection
All anomalies are interesting, and therefore all knowledge goals that arise during the anomaly detection process are interesting. These goals are always pursued in an attempt to form explanations to resolve the anomalies.

Knowledge goals that seek explanations in order to resolve anomalies are less interesting if explanations are easily available. If no explanation is found, there is a gap in memory corresponding to an unexplainable anomaly. A knowledge goal that seeks to fill in this gap is interesting for an understander that is trying to learn about this domain.

H-6: Explanation availability
An explanation retrieval goal that fails to find an explanation in memory is more interesting than one where an explanation can be easily found.

Thus stories that have standard explanations are less interesting than novel and unusual stories.

Of course, this and any other heuristic might be overruled if other interestingness heuristics came into play. Heuristics like these encode rules that represent guesses at interestingness, *other things being equal.* Counter-examples can easily be found by playing up interestingness factors represented by other heuristics. The final interestingness measure is a combination of all these factors, not any single factor taken by itself. For example, a stereotypical explanation for the motivations of a person who one knows personally might be more interesting than an unusual explanation for the motivations of a person one doesn't know. This also depends on the goals of the system, since a reasoner might be particularly interested in the motivations of a particular person for some other reason.

The interestingness of an explanation goal also depends on the kind of explanation that the goal is seeking. Knowledge goals seeking explanations for the motivations of individuals arise from *human interest stories*, and are usually more interesting, from the human interest point of view, than those that seek explanations for the motivations of institutions.

H-7: Human interest
Knowledge goals involving motivations of individuals are more interesting than those involving motivations of institutions.

Note that heuristic H-7 judges interestingness from a human interest point of view. Human interest stories focus on the goals, motivations and emotions of particular individuals. From the point of view of politics or counter-planning, however, the motivations of the institution might be more interesting than those of an individual, unless the individual was an important political figure in that institution.

When searching for explanations for actions, instrumental actions are less interesting than the actions that they are instrumental to. This is similar to the objective/means criterion of POLITICS that was described earlier.

H-8: Instrumentality
If an action is instrumental to or part of another action, the former action is less interesting than the latter. If an action has more than one action instrumental to it, or a MOP has more than one scene, the most interesting of the instrumental actions is that which is the goal scene of the MOP (or the "maincons" of a script).

Similarly, habitual or commonly performed actions are uninteresting by the following heuristic:

H-9: Thematic or stereotypical action
If the actor, or a group that the actor belongs to, is known to perform such actions or select such plans in service of a known thematic goal, the action is not interesting. Thus common plans for goals and routine thematic or occupation-related actions are uninteresting.

These heuristics are used to focus the understander's attention on the most interesting actions. Other actions are "explained away" by building simple explanations. For example, AQUA does not explain stereotypical actions in detail. The explanation it builds is simply "Because the actor often performs such actions in service of his goals," unless it has been unable to explain such actions in the past in which case it would have a pending question which would make this action interesting. Such heuristics allow AQUA to spend its time processing the more interesting aspects of the story. If these heuristics are absent, AQUA will still process the interesting aspects of the story, of course, but it will spend a lot more time processing uninteresting details as well.

5.2.2 Hypothesis formation. When a possible explanation is found, it is applied to the anomalous situation in order to construct a hypothesis that

[4] Since AQUA does not perform any text-level learning, text-level goals are not interesting to AQUA according to the learning criterion for interestingness.

might explain the anomaly. If a stereotypical explanation is available that applies easily and directly, the story conforms to the explanation that the understander already knows about, and is therefore not very interesting. On the other hand, if existing explanations do not apply to the situation, the story is novel and therefore interesting.

H-10: Hypothesis formation
If an available explanation applies easily and directly to the story, the story is not very interesting. More interesting is the case when a known explanation applies but leaves gaps which need to be filled in before the hypothesis is verified. The most interesting story is one in which known explanations do not fit the situation and need to be modified.

This heuristic follows from the claim that interestingness is a measure of what the understander might learn from processing the situation. Stories that identify gaps in the understander's memory are more interesting than those that fit into stereotypical molds that the understander already knows about, since the understander is unlikely to learn anything from processing the latter kind of stories.

5.2.3 Hypothesis verification. The final step in the explanation process is the verification of hypotheses. Facts in the story that are relevant to existing hypotheses are more interesting than facts that have no bearing on hypotheses currently in memory:

H-11: Hypothesis verification
An input fact is interesting if it helps to verify or refute a hypothesis that might explain an anomaly.

It is worth noting that these heuristics are dynamic and therefore a considerable improvement over static heuristics that select interesting features or facts on some arbitrary basis. For example, the color of an agent's hair is usually irrelevant in most stories. However, if this feature were statically marked as being uninteresting, an understander would be unable to correctly process a story in which this feature turned out to be interesting for some unforeseen reason.

5.3 Interestingness from memory goals

For an explanation-based program such as AQUA, explanation goals are more interesting than memory or text-level goals. However, memory-level tasks also give rise to heuristics for interestingness which, as before, are based on trying to identify gaps in memory which give the understander an opportunity to learn. For example, the basic learning mechanism in a program such as IPP [Lebowitz, 1980] is that of similarity-based generalization, a process that builds categories in memory by noticing similarities between instances or sub-categories and building generalizations based on these similarities. This theory of learning suggests the following heuristics:

H-12: Uniqueness
A category with a unique example or specialization is more interesting than one with several examples. If two categories have unique examples, the category higher in the type hierarchy is more interesting than the one that is lower down.

H-13: Symmetry
A category which lacks a symmetric category is more interesting than one for symmetric categories are known.

The symmetry can be along any of the dimensions used for similarity-based generalization in memory. For example, if the understander builds categories of occupations based on the gender of the actor, occupations in which both male and female actors are seen would be less interesting than those in which only males or only females are seen. This assumes, of course, that these categories play some functional role in achieving the overall goals of the understander, otherwise there would be no principled reason for either building the categories or judging their interestingness.

6 Computing interestingness by combining heuristics

We have presented a set of content- and structure-based heuristics for judging interestingness based on different types of understander goals. This set is not exhaustive, of course, but it illustrates the type of heuristics that an understander would use to determine interestingness and focus its attention. This allows the understanding process to be sensitive to the knowledge goals of the system.

The final measure of interestingness is derived by combining the recommendations of all the applicable heuristics. This is used to judge the interestingness of the system's knowledge goals, as well as the interestingness of facts that might be relevant to these knowledge goals.

H-14: To determine the interestingness of a fact or a knowledge goal, apply all the interestingness heuristics to the fact or knowledge goal and combine the interestingness recommendations of each heuristic.

There is a potential problem here since the heuristics given above don't recommend specific interestingness values. For example, is an anomaly involving an uninteresting goal more interesting than a stereotypical way of achieving a highly interesting goal? This problem has not yet been addressed. The current implementation of AQUA pursues every knowledge goal that is judged to be interesting by one or more heuristics.

7 Examples

In conclusion, let us illustrate the above interestingness heuristics by using them to determine the interestingness of some example stories from the terrorism domain. Consider the following story (New York Times, April 14, 1985):

S-1: Boy Says Lebanese Recruited Him as Car Bomber.

JERUSALEM, April 13 — A 16-year-old Lebanese was captured by Israeli troops hours before he was supposed to get into an explosive-laden car and go on a suicide bombing mission to blow up the Israeli Army headquarters in Lebanon. ...

What seems most striking about [Mohammed] Burro's account is that although he is a Shiite Moslem, he comes from a secular family background. He spent his free time not in prayer, he said, but riding his motorcycle and playing pinball. According to his account, he was not a fanatic who wanted to kill himself in the cause of Islam or anti-Zionism, but was recruited [by the Islamic Jihad] through another means: blackmail.

We can use the above heuristics to judge the interestingness of this story.

H-8 **Instrumentality:** The suicide bombing is not instrumental to a known larger plan.

H-3 **Vicarious crisis goal:** The boy's action affects his `preserve-life` goal.

H-6, H-10 **XP availability and applicability:** A stereotypical XP is available (religious fanatic) but inapplicable. Another stereotypical XP (blackmail) is applied in a novel context.

H-11 **Hypothesis verification:** The religious fanatic hypothesis is refuted. The blackmail explanation applies, but raises new questions such as "What could the boy want more than his own life?" which must be answered before the hypothesis is completely filled out.

H-7 **Human interest:** The explanation discusses personal motivation.

H-12 **Uniqueness:** The blackmail explanation has never been applied to a suicide bombing story before.

H-4 **Actor relevance:** The actor and planner are unknown to the understander.
Object relevance: The objects involved are unknown to the understander.

H-1 **Goal relevance:** No personal goals are achieved or violated.
Location relevance: Lebanon is not personally relevant to the understander.

This story, although not personally relevant, is interesting from the point of view of human interest. The story discusses novel explanations for the motivations behind the actions involving the violation of a shared thematic goal, `preserve-life`. Suppose AQUA has read several religious fanaticism stories, but has not encountered any coercion stories so far. After reading the above story, AQUA will be left with several questions, including:

- What did Mohammed value more than his own life?
- Why did the Islamic Jihad plan this mission?
- Why did the Islamic Jihad choose a teenager for the mission?

These questions are represented as knowledge goals in AQUA's memory. On the basis of these knowledge goals, AQUA will now be interested in stories involving the people or institutions it has questions about, such as:

- Another story about Mohammed Burro (relevance to known person)
- Another story about the Islamic Jihad (relevance to known institution)

AQUA will also be interested in stories involving the newly learned blackmail explanation, such as:

- Another story about someone being blackmailed into a suicide bombing mission (relevance to novel explanation)

AQUA will also be interested in a story involving goals or goal priorities that it has questions about, such as:

- Another story about someone valuing something over their own life (relevance to goal)

In contrast, consider the following more stereotypical story:

S-2: Suicide bomber strikes U.S. embassy in Beirut.

A teenage girl exploded a car bomb at the U.S. embassy in Beirut today, killing herself and causing a number of casualties, security sources said. A statement by an unidentified terrorist group claimed responsibility for the attack, adding that the girl was a martyr for the cause of Islam.

This story is relatively uninteresting. It is interesting only to the extent that it is about the U.S. embassy (personal relevance of object to an American understander), and that it discusses the motivations behind the violation of a vicarious crisis goal (heuristic H-3). However, since these motivations have a standard explanation that has been seen many times before (religious fanaticism), this story is not interesting even from the point of view of explanation (H-6, H-10).

8 Conclusion: Interest-producing conditions

We define interestingness as a criterion for inference control. Since the understander needs to focus its attention on those inferences likely to help it achieve its overall goals, it must devote its resources pursuing inferences that are most likely to be useful towards achieving these goals. Thus interestingness is a heuristic measure of the relevance of the input to the understander's knowledge goals. Since the point of satisfying knowledge goals is to improve one's understanding of the domain, interestingness can also be thought of as a measure of the likelihood of learning something from the story if one processes it in detail.

Interestingness is neither inherent in the information nor in the system, but rather arises from the interaction between the two. It arises from the interaction between the stimulus and the goals of the system. A system with no goals would have no reason to find any input more interesting than any other, nor would any particular piece of information be universally interesting for all systems unless they shared the same goals.

This is a functional approach to the problem of interestingness [Hidi and Baird, 1986; Schank, 1979] from the perspective of the theory of question-driven understanding. A similar approach can be used for systems performing other cognitive tasks, such as planning, since these systems would also need to focus their attention on inferences that were relevant to goals arising from their tasks.

In AQUA, interest in a concept is triggered by its likely relevance to questions or knowledge goals, and continuing interest is determined by its continuing significance to these goals. This is related to the "goal satisfaction principle" of [Hayes-Roth and Lesser, 1976], which states that more processing should be given to knowledge sources whose responses are most likely to satisfy processing goals, and to the "relevance principle" of [Sperber and Wilson, 1986], which states that humans pay attention only to information that seems relevant to them. These principles make sense because cognitive processes are geared to achieving a large cognitive effect for a small effort. To achieve this, the understander must focus its attention on what seems to it to be the most relevant information available [Sperber and Wilson, 1986].

Once the interestingness of a question or piece of input has been determined, AQUA uses it to guide processing by focussing its resources on the more interesting aspects of the story. Since the heuristics are geared towards learning, this ensures that AQUA spends its time on those aspects of the story that are most likely to result in something useful being learned. Without its interestingness heuristics, AQUA would still learn the same things, but it would spend a lot more time drawing inferences that ultimately turn out to be irrelevant.

References

[Carbonell, 1979] J. G. Carbonell. *Subjective Understanding: Computer Models of Belief Systems.* Ph.D. thesis, Yale Universtiy, New Haven, CT, January 1979. Research Report #150.

[Hayes-Roth and Lesser, 1976] F. Hayes-Roth and V. Lesser. Focus of attention in a distributed logic speech understanding system. In *Proceedings of the IEEE International Conference on ASSP*, Philadephia, PA, 1976.

[Hidi and Baird, 1986] S. Hidi and W. Baird. Interestingness — A Neglected Variable in Discourse Processing. *Cognitive Science*, 10:179–194, 1986.

[Lebowitz, 1980] M. Lebowitz. *Generalization and Memory in an Integrated Understanding System.* Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, October 1980. Research Report #186.

[Ram, 1987] A. Ram. AQUA: Asking Questions and Understanding Answers. In *Proceedings of the Sixth Annual National Conference on Artificial Intelligence*, pages 312–316, Seattle, WA, July 1987. American Association for Artificial Intelligence, Morgan Kaufman Publishers, Inc.

[Ram, 1989] A. Ram. *Question-driven understanding: An integrated theory of story understanding, memory and learning.* Ph.D. thesis, Yale University, New Haven, CT, May 1989. Research Report #710.

[Schank and Ram, 1988] R. C. Schank and A. Ram. Question-driven Parsing: A New Approach to Natural Language Understanding. *Journal of Japanese Society for Artificial Intelligence*, 3(3):260–270, May 1988.

[Schank, 1979] R. C. Schank. Interestingness: Controlling Inferences. *Artificial Intelligence*, 12:273–297, 1979.

[Sperber and Wilson, 1986] D. Sperber and D. Wilson. *Relevance: Communication and Cognition.* Language and Thought Series. Harvard University Press, Cambridge, MA, 1986.

The Dempster-Shafer Theory of Evidence as a Model of Human Decision Making

Donald H. Mitchell
Amoco Production Company
Tulsa Research Center
P.O. Box 3385
Tulsa, OK 74102*

Abstract

Many psychology researchers have shown that humans do not process probabilistic information in a manner consistent with Bayes' theory [9, 10, 16, 24, 23, 27]. Robinson and Hastie [24, 23] showed that humans made non-compensatory probability updates, produced super-additive distributions, and resuscitated zero probability possibilities. While most researchers have classified these behaviors as non-normative, we found that the Dempster-Shafer theory could model each of these behaviors in a normative and theoretically sound fashion. While not claiming that the theory models human processes, we claim that the similarities should aid user acceptance of Dempster-Shafer based decision systems.

1 Introduction

Due to the inherent uncertainty of evidence and conclusions in the world, decision support systems (including artificial intelligence systems) must often use methods for representing and reasoning under uncertainty. There are a number of possible methods. Each method has a different effect on the three major expert system stages: 1) acquisition, 2) inferencing, and 3) user interpretation of the results. While many products and papers downplay the importance, the choice is difficult and important. The chosen paradigm can mitigate or exacerbate errors in any of the stages thus making the system's results meaningless.

There are a number of results supporting each reasoning method. One attribute of comparison is theoretical soundness [6, 17, 18, 19, 2]. Most of these comparisons uphold the theoretical foundation of probability theory and particularly of Bayes' theorem. Another attribute is empirical performance [8, 1, 29, 22, 17, 20, 21]. These studies support a variety of conclusions. Dawes [8], for example, shows that using a simple, yet incorrect, linear model is often better than a theoretically sound probabilistic model when they are both based on the same error-prone human estimates. In this paper, we argue for a third attribute—user interpretation and acceptance.

When consulting a decision support system, a human's ability to understand the computer's beliefs and decisions is important. Early research on automated tools showed that users more readily accept systems if they understand the systems' behaviors [4, 5, 11, 14, 25]. This understanding can result from any of three processes: 1) training the human to understand the theoretical correctness of the reasoning processes, 2) using a reasoning process that directly corresponds to the human's or 3) using a process whose observable behavior corresponds to the human's. The first process is apt to meet with resistance and makes general distribution and acceptance difficult. The second process, while of great potential, is difficult to accomplish due to the hidden nature of human decision-making processes. The third process corresponds directly to the way that most collaborative human decision-making works: when humans defend their reasoning, they refer to the evidence that caused them to increase or decrease their belief and not to their reasoning mechanisms.

This paper uses experimentally observed similarities between the Dempster-Shafer theory of evidence [26] and humans solving a probabilistic updating task to argue that humans may more readily understand Dempster-Shafer based systems. The paper does not directly address user acceptance in that it does not involve actual users of a system, but it indirectly addresses acceptance through the ability of humans to empathize with the behavior.

*(918) 660-4270, dmitchell@trc.amoco.com

2 The task and human data

This paper reports an experiment that compared the behavior of an automated Dempster-Shafer evidence accumulation system with the behavior of humans performing the same evidence accumulation task. The task and human experimental results come from a study by Robinson and Hastie [24, 23].

In an effort to find out whether humans followed Bayesian probabilistic reasoning principles, Robinson and Hastie asked human subjects to solve a murder mystery. The subjects saw a series of clues. After each clue, the subjects stated their beliefs about the guilt of each suspect in terms of probability. Robinson and Hastie found that the humans did not follow probabilistic principles. We will explain the exact form of the discrepancy when we describe the behavior of the Dempster-Shafer system.

Before describing the Dempster-Shafer system, we will address two concerns with the Robinson and Hastie data. Some may argue that Robinson and Hastie's subjects did not have adequate training in the probabilistic concepts. To test this hypothesis, Robinson and Hastie explicitly taught the fundamentals of probability to some of the subjects. Depending on the subject, the training led to either the same behavior as those without training or a behavior that did not reflect any evidence accumulation. Robinson and Hastie conjectured that the cognitive overhead prevented the subjects from applying the learning.

Another objection may be that one study makes an insufficient basis for concluding that humans are not Bayesian decision makers. Robinson and Hastie, however, are not the only researchers to show that humans make poor Bayesian probabilistic information processors. Many other psychology researchers have shown the non-Bayesian character of human information processing [9, 10, 16, 24, 23, 27].

3 Dempster-Shafer predictions

To test the predictions of the Dempster-Shafer [26] theory, we developed a straight-forward implementation of the theory [20] and then submitted the clues to it as a series of consonant belief functions. Although many artificial intelligence researchers [3, 7, 13, 12] have restricted their Dempster-Shafer representations to simple and dichotomous belief functions, we chose consonant belief functions because they are the form that Shafer says most naturally represent inferential evidence [26, pp. 223–229]. The reason other artificial intelligence researchers have ignored this representation is that it can result in exponential computational time requirements. For the sake of this study, computational time was not of direct concern.[1]

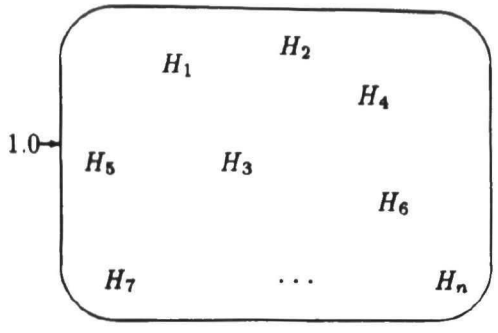


Figure 1: Venn graph of the vacuous function: there has been no evidence.

While it is not the intent of this paper to fully introduce the Dempster-Shafer theory, we will briefly describe belief functions. A belief function is the assignment of probabilities to sets of conclusions. This assignment differs from standard probabilistic theories in that it uses sets rather than single hypotheses. All probability theories can use sets to represent multiple simultaneously true hypotheses; however, the Dempster-Shafer system uses sets to indicate lack of differentiation in the evidence for mutually exclusive hypotheses. The interpretation of this assignment is that some element of the set is true but the evidence does not provide fine enough granularity to directly point to one hypothesis.

A consequence of this representation for belief is that there is a clear distinction between the inability to decide due to lack of evidence and the inability to decide due to too much conflicting evidence. In the Dempster-Shafer theory, a believer represents the lack of evidence as the assignment of all probability to the undifferentiated set of all possible hypotheses (e.g. figure 1), whereas the representation for conflicting evidence is the assignment of roughly equal amounts of probability to many separate singleton sets of hypotheses (e.g. figure 2). For example, in a well-matched musical competition, the judge's initial belief should be no one has evidence in their favor and all contestants can fight for the prize like a pie ready to be divided: there is no conflicting beliefs concerning the outcome. After listening to everyone perform

[1] Note, however, that [20, 15] both show linear complexity when applying the theory to naturally constrained problems.

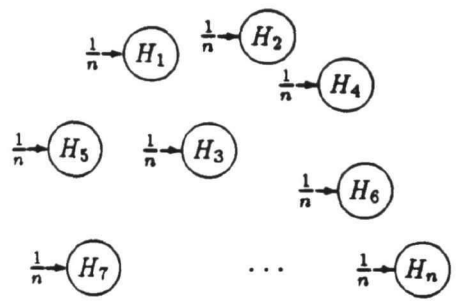


Figure 2: Venn graph of a maximally conflicting belief function: there has been evidence supporting each hypothesis equally.

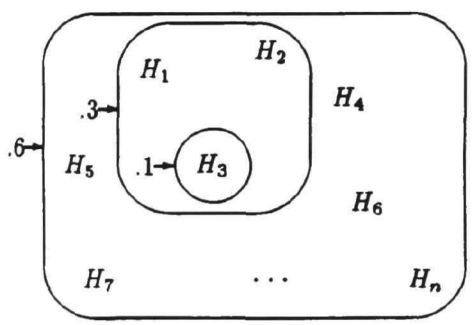


Figure 3: Venn graph of the consonant belief function: $\Pr(H_3) = .1$, $\Pr(H_1 \vee H_2 \vee H_3) = .3$, and $\Pr(H_1 \vee \cdots H_n) = .6$.

well, the judge should have belief in each contestant individually with little or no residual belief: each contestant has claims to more pie than is available. The evidence causes conflicting beliefs.

More generally, assigning non-zero probabilities to sets of hypotheses that don't subsume one-another represents conflicting belief. The common Dempster-Shafer representation used in artificial intelligence of assigning belief to a hypothesis and its negation, thus, directly encodes conflict. Individual pieces of evidence, however, should not show any conflict with themselves, and, therefore, this dichotomous representation is usually inaccurate. The consonant belief function is the non-conflicting alternative.

Figure 3 depicts a consonant belief function. In this example, some evidence supports hypothesis H_3. The same evidence has less direct support for hypotheses H_1 and H_2 and also has some residual uncertainty. This belief function is represented by assigning non-zero probabilities to the sets H_3, H_1, H_2, H_3, and finally to $H_1 \ldots H_n$. This gradual focusing of probabilities on progressive subsets is the definition of a consonant belief function. A consonant belief function is consonant with itself: that is, it shows no conflict with itself.

The consonance of an individual piece of evidence with itself implies nothing about consonance between pieces of evidence. Different pieces of evidence can conflict with each other. The detective story used in this experiment, for example, shows considerable conflict between clues. While each clue might be self-consistent and therefore consonant, that does not imply that all clues agree. The result of combining these disagreeing but self-consonant belief functions will not be consonant.

To make a choice among the hypotheses requires comparing the beliefs assigned to each hypothesis. In the Dempster-Shafer theory, there is not a single measure of belief for individual hypotheses. Shafer provides several measures. The most important ones are the Bel function that indicates the lower bound of belief and the Pl plausibility function that indicates the upper bound. This experiment uses both of these to compare the Dempster-Shafer system's results with the human subjects' guilt estimates.

4 Comparisons of behavior

One way in which the humans did not follow probabilistic principles was that they usually changed only the probability of the suspect directly impugned by the clue without making compensatory changes to the other suspects. Because probability requires that the sum of the probabilities over suspects equals 1.0, each change must be balanced with an equal change in the opposite direction for the other suspects. Bayesian probability requires proportionately equal changes in the non-impugned hypotheses. Robinson and Hastie termed their subjects' omission of this required compensation "non-compensatory probability updating."

Although subjects in general did not compensate, there were two conditions under which they did, at least partially: 1) when a clue had an extreme impact on the guilt of one suspect, and 2) after a large number of clues had already been processed. In the first case—extreme impact—the clue often contradicted prior belief: that is, it indicated that a subject's favorite suspect was actually innocent or that

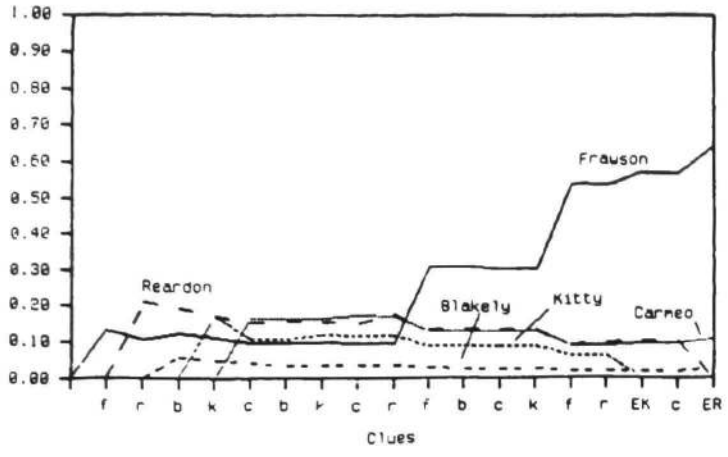


Figure 4: Dempster-Shafer Bel score guilt ratings for each suspect after each clue.

a long-shot was guilty.

Figure 4 shows the Dempster-Shafer system's Bel function assignment of guilt to each suspect after each clue. Each point on the abscissa is a separate clue. The first initial of the suspect mentioned by the clue marks each point. Some of the clues indicate guilt, some indicate innocence, and some are neutral. Like the human subjects, most of the Dempster-Shafer system's belief changes were non-compensatory. The only strong demonstration of compensation occurred in reaction to the two strong clues regarding Frawson and the clues eliminating Reardon and Kitty. This behavior corresponds exactly with Robinson and Hastie's descriptions of their subjects.

As an explanation for the humans' behavior, this result suggests that the humans may not fully partition their belief before collecting evidence. The humans may take an approach that is analogous to the Dempster-Shafer system's approach: that is, slowly portion out belief and only when there is a preponderance of evidence for one hypothesis do they take back belief from other hypotheses. We are not, however, claiming that the Dempster-Shafer system literally models the individual subjects. There was far too much variance between subjects to even attempt to analyze the system's ability to model the individuals.

Robinson and Hastie found two other aspects of the human data that conflicted with probability: probabilities usually added to over one—"super-additivity"—and some humans sometimes gave non-zero probability ratings to suspects after giving them zero ratings—"resuscitation."

If the Dempster-Shafer upper-bound probability measure Pl is used, then most of the probabilities add to over one thus qualitatively modeling the super-additivity. The interpretation in this case is that the subjects were sensitive to their residual uncertainty about the suspects and felt that the lower-bound estimates made them look overly convinced of the suspect's innocence. Assigning probabilities as low as the Bel scores in figure 4 might look like an admission of implausibility that did not correspond to the subjects' beliefs. The scoring mechanism did not give the subjects any way to indicate the suspects' potential guilt, and, therefore, the subjects may have blended the plausibility score with the actual belief.

Because the Bel measure naturally increases from zero to some non-zero value as evidence is collected, the Bel score could model the resuscitation. To use this explanation in conjunction with the super-additivity explanation requires the assumption that the subjects were somehow sensitive to both the Bel and Pl and chose to respond in some hybrid manner that sometimes allowed the Bel score to override the Pl score.

Simultaneously using the Bel and Pl measures to explain the human behavior is not adequately convincing especially because there are other possible explanations for the super-additivity and resuscitation. Explanations based on the input scale and human understanding seem more appealing than an explanation based on the Dempster-Shafer theory. For example, there are some problems with Robinson and Hastie's method of soliciting probability ratings. They used a scale marked into 0.05 probability intervals. The subjects may not have realized that position within an interval was significant. This explanation could explain the resuscitation effect because no suspect resuscitated to more than a probability of 0.1.

While the Dempster-Shafer system does provide an explanation of these anomalies, we feel that the major contribution of this work is to propose a behavior with which decision system users could empathize. The similarity is especially strong for the non-compensatory behavior. Because the Dempster-Shafer system is theoretically sound, system developers can feel secure using it.

5 Conclusion

If, as previous work has suggested [4, 11, 14, 28, 25], user acceptance depends on the ability of the user to empathize with system behavior, and if humans are particularly poor at understanding Bayesian probabilistic notions, then this result showing the similarity between human and Dempster-Shafer updating behaviors encourages further exploration of the

use of the Dempster-Shafer theory in automated reasoning. These results combined with the Dempster-Shafer theory's theoretical soundness and Mitchell's results [20, 21] concerning acquisition and computational requirements are a strong argument in favor of the Dempster-Shafer theory.

References

[1] Tversky A. and D. Kahneman. Causal schemas in judgments under uncertainty. In M. Fishbein, editor, *Progress in Social Psychology*, volume 1, pages 49–72. Lawrence Erlebaum, Hillsdale, NJ, 1980.

[2] AAAI. *Uncertainty in Artificial Intelligence*, 1987.

[3] J. A. Barnett. Computational methods for a mathematical theory of evidence. In *International Joint Conference on Artificial Intelligence*, pages 868–875, Vancouver, British Columbia, Canada, 1981.

[4] H. L. Bleich. The computer as a consultant. *New England Journal of Medicine*, 284(3):141–147, 1971.

[5] B. G. Buchanan and E. H. Shortliffe. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley, Reading, MA, 1984.

[6] P. Cheeseman. In defense of probability. In *International Joint Conference on Artificial Intelligence*, pages 1002–1009, 1985.

[7] Bruce D'Ambrosio. A hybrid approach to reasoning under uncertainty. *International Journal of Approximate Reasoning*, 2:29–45, 1988.

[8] R. Dawes. The robust beauty of improper linear models in decision making. *American Psychologist*, 34:571–582, 1979.

[9] Ward Edwards. Conservatism in human information processing. *Nature*, 32:414–416, 1971.

[10] B. Fischhoff and R. Beyth-Marom. Hypothesis evaluation from a Bayesian perspective. *Psychological Review*, 90(3):239–260, 1983.

[11] R. B. Friedman and D. H. Gustafson. Computers in clinical medicine, a critical review. *Computers and Biomedical Research*, 10:199–204, 1977.

[12] J. Gordon and E. Shortliffe. A method for managing evidential reasoning in a hierarchical hypothesis space. *Artificial Intelligence*, 26:323–357, 1985.

[13] J. Gordon and E. H. Shortliffe. The Dempster-Shafer theory of evidence. In B. G. Buchanan and E. H. Shortliffe, editors, *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, pages 272–292. Addison-Wesley, Reading, MA, 1984.

[14] G. A. Gorry. Computer-assisted clinical decision-making. *Meth. Inform. Med.*, 12(1):45–51, 1973.

[15] R. C. Hughes and J. N. Maksym. Acoustic signal interpretations: Reasoning with non-specific and uncertain information. *Pattern Recognition*, 18(6):475–483, 1985.

[16] Daniel Kahneman, Paul Slovic, and Amos Tversky, editors. *Judgment Under Uncertainty: Heuristics and Biases*. Cambridge University Press, New York, 1982.

[17] Jin H. Kim and J. Pearl. A computational model for causal and diagnostic reasoning in inference systems. In *International Joint Conference on Artificial Intelligence*, pages 190–193, 1983.

[18] Lemmer and Kanal, editors. *Uncertainty in Artificial Intelligence: the 1985 workshop*. AAAI, North-Holland, 1985.

[19] Lemmer and Kanal, editors. *Uncertainty in Artificial Intelligence: the 1986 workshop*. AAAI, North-Holland, 1986.

[20] Donald H. Mitchell. *Automated Decision Support Using Variations on the Dempster-Shafer Theory*. PhD thesis, Northwestern University, 1987.

[21] Donald H. Mitchell, Steven A. Harp, and David K. Simkin. A knowledge-engineer's comparison of three evidence aggregation methods. In *Second Workshop on Uncertainty in AI*, pages 297–304. AAAI, Morgan Kauffman, 1987.

[22] J. Pearl. On evidential reasoning in a hierarchy of hypotheses. *Artificial Intelligence*, 28:9–15, 1986.

[23] L. B. Robinson. *In the Footsteps of Sherlock Holmes: Information Search and Hypothesis Testing*. PhD thesis, Northwestern University, 1986.

[24] L. B. Robinson and R. Hastie. Revision of beliefs when a hypothesis is eliminated from consideration. *Journal of Experimental Psychology: Human Perception and Performance*, 11(4):443–456, 1985.

[25] William B. Rouse and Nancy M. Morris. Understanding and enhancing user acceptance o computer technology. *IEEE Transactions on Systems, Man, and Cybernetics*, 16(6):965–973, 1986.

[26] Glenn Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[27] P. Slovic and S. Lichtenstein. Comparison of Bayesian and regression approaches to the study of information processing judgement. *Organizational Behavior and Human Performance*, 6:649–744, 1971.

[28] T. S. Startsman and R. E. Robinson. The attitudes of medical and paramedical personnel toward computers. *Computers and Biomedical Research*, 5:218–227, 1972.

[29] A. Tversky and D. Kahneman. Causal schemas in judgments under uncertainty. In Kahneman, Slovic, and Tversky, editors, *Judgment Under Uncertainty: Heuristics and Biases*, pages 117–128. Cambridge University Press, 1982.

Feature Selection and Hypothesis Selection Models of Induction

Michael J. Pazzani & Glenn Silverstein
Department of Information and Computer Science
University of California
Irvine, CA 92717

Abstract

Recent research has shown that the prior knowledge of the learner influences both how quickly a concept is learned and the types of generalizations that a learner produces. We investigate two learning frameworks that have been proposed to account for these findings. Here, we contrast *feature selection* models of learning with *hypothesis selection* models. We report on an experiment that suggests that human learners use prior knowledge both to indicate what features may be relevant and to influence how the features are combined to form hypotheses. We present an extension to the POSTHOC system, a hypothesis selection model of concept learning, that is able to account for differences in learning rates observed in the experiment.

Introduction

There is a growing body of evidence that the prior knowledge of the learner influences the speed or accuracy of learning (e.g., Ahn, Mooney, Brewer, & DeJong, 1987; Ausubel & Schiff, 1954; Chapman & Chapman, 1967; Murphy & Medin, 1985; Nakamura, G, 1985; Pazzani, 1990; Schank, Collins, & Hunter, 1986; Wattenmaker, Dewey, Murphy, & Medin, 1986; Wisniewski,1989). In this paper, we contrast two learning frameworks that have been proposed to account for these findings. *Feature selection* models, such as that proposed by Lien & Cheng (1989), claim that prior knowledge influences learning by selecting a subset of the available features as potentially relevant. In the feature selection model, induction is accomplished by detection of covariation (Kelley, 1971; 1983) among the relevant features. In contrast, *hypothesis selection* models, exemplified by POSTHOC (Pazzani & Schulenburg,1989), claim that, in addition to selecting relevant features, prior knowledge influences how the relevant features are combined to form hypotheses and how hypotheses are revised when new data are encountered.

In this paper, we first present an experiment in which the feature selection and hypothesis selection frameworks make different predictions on learning rates. Next, we report on an extension to POSTHOC that provides it with the capability to reason about both positive and negative causal influences (i.e., factors that make an action more likely or less likely). Finally, we report on a simulation that indicates that POSTHOC can account for the learning rates observed in the experiment.

Drug Interactions: An Experiment

In order to differentiate between these two frameworks, we designed an experiment in which the their predictions differed. The experiment followed a 2x2 factorial design in which the factors were the type of concept acquired (internal or external disjunction) and the type of background information in the instructions. It has been reported (Wells, 1963), that in the absence of relevant prior knowledge, exclusive disjunctions concepts are more difficult for subjects to learn that inclusive disjunctive concepts. The feature selection model would predict that an exclusive disjunction of relevant features would take more trials to learn than an inclusive disjunction of the same relevant features. The reason for this prediction is that the feature selection model predicts that prior knowledge will influence only the selection of features. After this selection is made, one would expect the same degree of difficulty as in the case in which subjects have no relevant prior knowledge. In contrast, the hypothesis selection model predicts that an exclusive disjunction may be easier for subjects to learn than an inclusive disjunction, if the exclusive

disjunction of relevant features is consistent with the subjects prior knowledge, but the inclusive disjunction of these same features is not consistent.

In the experiment, subjects were told that they were to review records of patients brought to the emergency room of a hospital because of an overdose of sleeping pills and that the effect of the sleeping pills was to lower the patients heart rate. The patient records were described by the following five features and that each feature had one of two values:

• Gender:	The gender of the patient.	(Female or Male)
• Time:	The time of day.	(AM or PM)
• Oral:	Each patient is given a capsule to swallow.	(Drug-o or Sugar)
• Intravenous:	Each patient is given an injection.	(Drug-i or Saline)
• Doctor:	The attending physician.	(Ramsey or Jankins)

The subjects had to learn a way to predict whether or not the patient's heart rate will increase. Subjects had to learn either an inclusive or an exclusive disjunction of Drug-o and Drug-i (i.e., some subjects were presented with data that indicated that a patient's heart rate would increase only if a patient was given either Drug-i or Drug-o; other subjects were presented with data that indicated that the heart rate would increase only if a patient was given Drug-i or Drug-o, but not both).

The instructions also included background information. Two sets of instructions were prepared. The only difference between them is that one set of instructions omitted the item underlined below:

- The nurses in the PM shift receive a 10% higher salary than those in the AM shift.
- Female patients typically weigh less than male patients.
- Drug-o has been used by truck drivers to stay alert.
- Drug-i has been shown to increase aggressiveness in primate studies.
- When both Drug-o and Drug-i are given to laboratory animals, they result in a coma.
- Dr. Ramsey received his degree from Rutgers University in 1959.
- Dr. Jankins received his degree from Yale University in 1988.

Note that the background information contained items irrelevant to this particular task and that the relevant background information required plausible reasoning rather than purely deductive reasoning.

The hypothesis selection model makes several predictions about the outcome of the experiment. The predictions all follow from the thesis that concepts consistent with prior knowledge take fewer trials to learn than concepts that are not consistent with prior knowledge:

- Subjects learning the exclusive disjunction who were shown information on the drug interaction would learn more rapidly than subjects learning this concept without this information.

- Subjects learning the exclusive disjunction who were shown information on the drug interaction would learn more rapidly than subjects learning an inclusive disjunction with this information.

- Subjects learning the inclusive disjunction who were not shown information on the drug interaction would learn more rapidly than subjects learning this concept who were shown this information.

In contrast, the feature selection model would predict that including or omitting the information on the interaction between Drug-i and Drug-o would not affect the learning rate. It assumes that prior knowledge only focuses attention on features and covariation alone indicates how the features are combined.

Subjects. The subjects were 52 male and female undergraduates attending the University of California, Irvine who participated in this experiment to receive extra credit in an introductory psychology course. Subjects were randomly assigned to one of the four conditions. Subjects were tested in two groups of 26.

Stimuli. The stimuli consisted of patient records that were displayed on the monitor of a Macintosh computer. Since there are five two-valued features, a total of 32 records were constructed.

Procedures. Each subject was shown a patient record on the computer screen and asked to predict whether or not the heart rate would increase by clicking on a box containing the word Yes or a box containing the word No (i.e., using a mouse to move a pointer to the box and pressing a button on the mouse) . While still displaying the patient record, the computer indicated the correct answer by displaying the word Increase or Decrease. Next, the subject clicked on a box labeled Continue and the next patient record was shown. This process was repeated until the subjects were able to predict or classify correctly on 7 consecutive trials. The subjects were allowed as much time as they wanted to make their prediction and to view the record after the correct answer was shown. We recorded the number of the last trial on which the subject made an error. The records were presented in a random order. If the subject did not obtain the correct answer after 60 trials, we recorded that the last error was made on trial 60. The subjects were permitted to consult the instructions, containing information on operating the computer and background information at any time during the experiment.

Results. Table 1 displays the results of the four conditions. The results of this experiment confirmed the predictions of the hypothesis selection model ($p < .05$, level $F(1, 48) = 4.48$). A Tukey HSD finds a significant difference ($p < .05$) on the the following comparisons:

- Subjects learning an exclusive disjunction and provided with information on the drug interaction did learn more rapidly than subjects learning the same concept without this background knowledge (7.3 vs. 28.2). This difference suggests that the knowledge of the interaction between the drugs facilitates learning this concept.

- Subjects learning an exclusive disjunction and provided with information on the drug interaction would learn more rapidly than subjects learning an inclusive disjunction and provided with information on drug interaction. (7.3 vs. 16.2). In this case, the knowledge of the interaction interferes with learning an inclusive disjunction since this hypothesis is not consistent with the prior knowledge.

Although not statistically significant, the data do not contradict the third prediction of the hypothesis selection framework:

- Subjects learning an inclusive disjunction and provided with information on the drug interaction would learn less rapidly than subjects learning an inclusive disjunction without this extra misleading knowledge (14.9 vs. 16.2). If we ignore the score of a subject who failed to complete the experiment, this result is more in line with our expectations (11.3 vs. 16.2).

Table 1. Mean number of trials required by human subjects

	Inclusive	Exclusive
With knowledge of interaction	16.2	7.3
Without knowledge of interaction	14.9	28.2

The results of this experiment provide support for hypothesis selection models of concept learning. In this framework, the hypothesis space is reduced by eliminating those hypotheses that are not consistent with the prior knowledge of the learner. In contrast, feature selection models restrict the hypothesis space less than hypothesis selection models. All hypotheses composed of potential relevant features are considered consistent with prior knowledge. As a consequence, the feature selection framework does not account for the differences in learning rates observed in this experiment.

POSTHOC: A hypothesis selection model

POSTHOC (Pazzani & Schulenburg, 1989) is a hypothesis selection model of concept learning. Through the use of a set of productions and a background theory that represents prior knowledge, POSTHOC maintains a single hypothesis that summarizes the examples seen and classifies new examples. The productions are used to suggest hypotheses and to revise hypotheses that misclassify examples. Because some of the productions do not make use of background knowledge, the system has the ability to create hypotheses that are not consistent with its background knowledge (e.g., if the background knowledge is incomplete or incorrect). In this paper, we discuss an extension to the system that allows it to make use of negative as well as positive influences. Note that POSTHOC was not modified in anyway to make it learn exclusive disjunctions. Rather, hypotheses representing exclusive disjunctions are formed using background knowledge when there are two separate features that positively influence a result, but the combination of the features negatively influence the result. Without the background knowledge of the specific drug interaction, POSTHOC would create an initial hypothesis consistent with its theory and latter be forced to revise them using productions that ignore the background knowledge. If POSTHOC has no background theory at all, then it would create its hypotheses using only productions that ignore background knowledge.

In POSTHOC, examples are expressed as set of feature-value pairs and an outcome. For example, in the medical experiments described in the previous section, a patient record describing a male patient who was administered Drug-o orally and a saline solution intravenously after noon and whose heart rate increased would be represented as follows:

[gender male] [time pm] [oral drug-o] [intravenous saline] $\in$ increase

POSTHOC maintains a single hypothesis that consists of a DNF description (i.e., disjunction of conjunctions) of the concept being learned. One hypothesis that is consistent with the above example states that the heart rate will increase if Drug-o is given orally in the afternoon.

[oral drug-o] $\wedge$ [time pm] $\rightarrow$ increase

Of course, there are numerous other hypotheses consistent with this example that may or may not be consistent with future examples or with the prior knowledge of the learner.

The influence theory which comprises POSTHOC's prior knowledge consists of two components: a set of influences which describe tendencies that either facilitate or hinder the desired outcome (e.g. increasing the heart rate) and a set of inferences rules that indicate when these influences are present. The influence theory of POSTHOC for the medical experiment described in the previous section includes the following influences and inferences:

Influences:

```
(easier more-alert  increase)
(easier more-aggressive  increase)
```

Inferences:

```
(implies [oral drug-o]  more-alert)
(implies [intravenous drug-i] more-aggressive)
```

These influences and associated inferences suggest that making a person more alert or more aggressive can facilitate increasing that persons heart rate and taking Drug-o orally tends to make a person more alert and taking Drug-i intravenously tends to make a person more aggressive.

The influence theory described above includes only positive influences. However, some influences can only be expressed naturally as negative influences. One such example the knowledge of drug interactions which arises in the medical experiment described in the previous section. For instance, if we wished to include the knowledge that Drug-i and Drug-o tend to interact negatively in the patient to produce a coma (thus lowering the heart rate), then this knowledge is represented as the following influence and associated inference:

Influences:

```
(harder coma  increase)
```

Inference

```
(implies ([oral drug-o] ∧ [intravenous drug-i]) coma)
```

Adding negative influences to POSTHOC required extending the productions presented in Pazzani & Schulenburg (1989). There are three types of productions. One set deals with errors of commission in which a positive example is falsely classified as a negative example. These productions makes the hypothesis more general. The second set deals with errors of omission in which a negative example is falsely classified as a positive example. These productions makes the hypothesis more specific. The final set creates an initial hypothesis when the first positive example is encountered. For brevity only those productions which utilize the negative influences will be described. For a description of the remaining productions, the reader is referred to Pazzani & Schulenburg (1989):

Initializing Hypothesis.

```
IF there are features of the example that are indicative of the
inverse of a negative influence
THEN initialize the hypothesis to the negation of the conditions
indicative of the negative influence.
```

Errors of Omission.

```
IF the hypothesis is consistent with the influence theory
AND there are features that are indicative of the inverse of a
negative influence
THEN create a conjunction of the current hypothesis and the negation
of the conditions indicative of the negative influence.
```

Errors of Commission.

```
IF the  hypothesis is consistent with the background theory
AND for each true conjunction there are features not present in the
current example that would be necessary for the inverse of a negative
influence
THEN modify the conjunct by conjoining the negation of the conditions
indicative of the negative influence.
```

To illustrate the use of these productions, a trace of POSTHOC learning an exclusive disjunction is provided below. For brevity, we omit the "doctor" attribute from the examples. The first example that POSTHOC is presented with is an example of a treatment that successfully increases the patient's heart rate where a male patient is administered a sugar pill orally and Drug-i intravenously in the PM by Dr. Ramsey:

[gender male] [time pm] [oral sugar] [intravenous drug-i] ∈ increase

Since there is no initial hypothesis, POSTHOC uses an initialization production to create a hypothesis that accounts for the outcome of this example. A positive influence more-aggressive is present and POSTHOC creates the hypothesis that Drug-i leads to the increased heart rate:

[intravenous drug-i] → increase

This hypothesis is consistent with several more examples. Next, an example is presented where a patient is administered Drug-o orally and a saline solution intravenously. Here an error of omission occurs since POSTHOC predicts that the patient's heart rate will not increase but it does increase. The example encountered is:

[gender male] [time am] [oral drug-o] [intravenous saline] ∈ increase

The hypothesis is revised by an Error of Omission production for positive influences a multiple sufficient hypothesis is produced:

[intravenous drug-i] ∨ [oral drug-o] → increase

Again this hypothesis is consistent with several more examples. However, POSTHOC makes the wrong prediction when it encounters an example where the patient is administered both Drug-i intravenously and Drug-o orally. POSTHOC predicts that the patient's heart rate will increase, but the patients heart rate does not increase. This results in an error of commission. The example presented to POSTHOC is:

[gender male] [time am] [oral drug-o] [intravenous drug-i] ∉ increase

To correct its hypothesis, POSTHOC uses the Error of Commission production for negative influences. For the each disjunct, [oral drug-o] and [intravenous drug-i], the negation of the features indicative of the negative influence is conjoined with each conjunct, the results hypothesis is:

[intravenous drug-i] ∧ **not** ([oral drug-o] ∧ [intravenous drug-i]) ∨
[oral drug-o] ∧ **not** ([oral drug-o] ∧ [intravenous drug-i]) → increase

This hypothesis can be simplified to:

[intravenous drug-i] ∧ **not** ([oral drug-o]) ∨
[oral drug-o] ∧ **not** ([intravenous drug-i]) → increase

which is consistent with the remaining examples and represents the exclusive disjunction.

Simulation Results

We ran POSTHOC on 200 random orderings of the data on each of the same four conditions used in the experiment on human subjects. The results are shown in Table 2. The negative influence (i.e., the drug interaction) was not used to simulate the condition in which this item was not included in the instructions. The data show the same trends as the human experimental data: learning the exclusive disjunction of administering Drug-i intravenously and Drug-o orally was facilitated by the knowledge that Drug-i and Drug-o interact to put the patient in a coma (8.3 vs. 45.7) ; when provided with information on the drug interaction, learning the exclusive disjunction of the drugs was easier than learning the inclusive disjunction (6.0 vs. 8.3); learning an inclusive disjunction of the drugs when provided with misleading information on the drug interaction required more trials than the same concept without this extra misleading knowledge (6.0 vs. 3.7).

Table 2. Mean number of trials required by POSTHOC

	Inclusive	Exclusive
With knowledge of interaction	6.0	8.3
Without knowledge of interaction	3.7	45.7

In POSTHOC, we have focused on how prior knowledge influences learning rates and we have so far ignored other information used by human learners (e.g., perceptual salience of features, Bower & Trabasso, 1968). As a consequence, POSTHOC is not intended to make quantitative predictions on the number of training examples but rather predicts the relative difficulty of learning.

Future Directions

There are three possible direction in which we plan to extend the hypothesis selection model. First, we would like to be able to use the prior knowledge of the learner to influence the interpretation of ambiguous feature (Medin & Wisniewski, 1990). Second, we would like POSTHOC to be able to use more abstract knowledge. Currently, POSTHOC can represent the information that there there is a specific interaction between two drugs or that there is no drug interaction. In contrast, our subjects also appeared to have more general knowledge that indicates such things as drugs may interact and can use this knowledge to explain the specific interaction seen in the experiment in terms of the general knowledge of drug interactions. Finally, we plan to extend POSTHOC so that when it learns accurate hypotheses that are not consistent with its background knowledge, the background knowledge is revised to accommodate the new findings.

Conclusions

We have presented experimental evidence that provides support for hypothesis selection models of concept learning. We have extended POSTHOC to include negative influences and shown that with this extension alone, it is able to predict the relative order of difficulty of trials on inclusive and exclusive disjunctions. Recent work on the analysis of the limitations of inductive learning algorithms (Valiant, 1984; Dietterich, 1989) is in sharp contrast to the versatility demonstrated by human learners. We believe that approaches that make use of background knowledge to focus all aspects of learning are central to accounting for the generality of human learning.

Acknowledgements

The software used to run the experiment was designed by Francis Nguyen and Takeshi Tsubota. We would like to thank Kamal Ali, Cliff Brunk, David Foster, and Scott Truesdel for assistance in running the experiment and Gupi Silverstein and Caroline Ehrlich for commenting on an earlier draft of the paper. This research is supported in part by National Science Foundation Grant IRI-8908260.

Bibliography

Ahn, W., Mooney, R., Brewer, W., & DeJong, G. (1987). Schema acquisition from one example: Psychological evidence for explanation-based learning. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society.* Seattle, WA: Lawrence Erlbaum Associates.

Ausubel, D. M., & Schiff, H. M. (1954). The effect of incidental and experimentally induced experience on the learning of relevant and irrelevant causal relationships by children. *Journal of Genetic Psychology, 84,* 109-123.

Bower, G., & Trabasso, T. (1968). *Attention in learning: Theory and research.* New York: John Wiley and Sons.

Chapman, L. J., & Chapman, J. P. (1967). Genesis of popular but erroneous diagnostic observations. *Journal of Abnormal Psychology, 72*, 193-204.

Dietterich, T. (1989). Limitations on inductive learning. *Proceedings of the Sixth International Workshop on Machine Learning* (pp. 124-128). Ithaca, NY: Morgan Kaufmann.

Kelley, H. (1971). Causal schemata and the attribution process. In E. Jones, D. Kanouse, H. Kelley, N. Nisbett, S. Valins & B. Weiner (Eds.), *Attribution: Perceiving the causes of behavior*. Morristown, NJ: General Learning Press.

Kelley, H. (1983). The process of causal attribution. *America Psychologist*, 107-128.

Lien, Y., & Cheng, P. (1989). A framework for psychological induction: Integrating the power law and covariation views. *The Eleventh Annual Conference of the Cognitive Science Society.* (pp. 729-733).Ann Arbor, MI: Lawrence Erlbaum Associates, Inc.

Medin, D., & Wisniewski, E. (1990). Paper presented at the Symposium on Computational Approaches to Concept Formation, Stanford, CA.

Murphy, G., & Medin, D. (1985). The role of theories in conceptual coherence. *Psychology Review, 92*, 289-316.

Nakamura, G. (1985). Knowledge-based classification of ill-defined categories. *Memory & Cognition. 13*, 377-384.

Pazzani, M. & Schulenburg, D. (1989). The influence of prior theories on the ease on concept acquisition. *The Eleventh Annual Conference of the Cognitive Science Society.* (pp. 812-819).Ann Arbor, MI: Lawrence Erlbaum Associates, Inc.

Pazzani, M. (1990). *Creating a memory of causal relationships: An integration of empirical and explanation-based learning methods.* Hillsdale, NJ : Lawrence Erlbaum Associates.

Schank, R., Collins, G., & Hunter, L. (1986). Transcending inductive category formation in learning. *Behavioral and Brain Sciences, 9*, 639-686.

Valiant, L. (1984). A theory of the learnable. *Communications of the Association of Computing Machinery, 27*, 1134-1142.

Wattenmaker, W., Dewey, G., Murphy, T., & Medin, D. (1986). Linear severability and concept learning: Context, Relational properties and concept naturalness. *Cognitive Psychology, 18*, 158-194.

Wells, H. (1963). Effects of transfer and problem structure in disjunctive concept formation. *Journal of Experimental Psychology, 65*, 63-69.

Wisniewski, E. (1989). Learning from examples: The effect of different conceptual roles. *The Eleventh Annual Conference of the Cognitive Science Society.* (pp. 980-986).Ann Arbor, MI: Lawrence Erlbaum Associates, Inc.

A Rule Based Model of Judging Harm-doing[1]

Thomas R. Shultz

McGill University

ABSTRACT

A rule based computational model of the judgment of harm-doing is presented that qualitatively simulates the major principles of an emerging psychological theory of common sense moral reasoning. Simulation results indicate that the model, called MR for Moral Reasoner, generates verdicts in substantial agreement with those reached in somewhat difficult court cases. A higher rate of agreement with outcomes produced in simpler cases from traditional cultures suggests that the model possesses a good deal of cultural universality. Systematic damaging of the rules in the model indicated that most of the rules are essential in producing a high rate of agreement with court decisions and identified some rules regarding the mental state of the accused that, individually, are less essential because they compensate for each other.

A PSYCHOLGICAL THEORY OF JUDGING HARM-DOING

This project concerns the common sense evaluation of harm-doing. Whenever a person may have been harmed by someone else, a number of issues naturally arise. How was the harm caused, is anyone responsible for the harm, is that person blameworthy, and how much should he be punished? People encounter such cases of harm-doing frequently, either directly or through secondary accounts.

Shultz and Schleifer (1983) have developed a psychological theory of reasoning about harm-doing that was inspired principally by conceptual analyses in jurisprudence and moral philosophy. A brief synopsis of this theory is presented here, minus the philosophical motivation and supporting psychological evidence, much of which is reviewed in Darley and Shultz (1990). The main concepts and decisions in the theory are illustrated in Figure 1. For a case in which a person may have done something to harm someone else, major decisions focus on causation, morally responsibility, blame, and punishment. Each of these major decisions presupposes and uses information from previous major decisions.

Judgments of moral responsibility, for example, presuppose those of causation. If the accused is judged not to have caused the harm, then there is no need to consider whether he is morally responsible for it. Similarly, judgments of blame presuppose those of moral responsibility, and decisions about punishment presuppose those about blame. A person is responsible for harm that he caused if the harm cannot be excused. Blame refers to a decision that a person is at fault, given that he has caused and is responsible for the harm. Without responsibility, there is no need to consider blame. Punishment refers to a decision about what consequences should befall the person as a result of being blameworthy. If the person is blameless, then no decision needs to be taken about punishment.

---

[1] I am grateful to John Darley and Kevin Dunbar for helpful comments on this research. This research is supported by a grant from the Social Sciences and Humanities Research Council of Canada. Address correspondence to Thomas R. Shultz, Department of Psychology, McGill University, 1205 Penfield Avenue, Montréal, Québec, Canada H3A 1B1. E-mail: ints@musicb.mcgill.ca

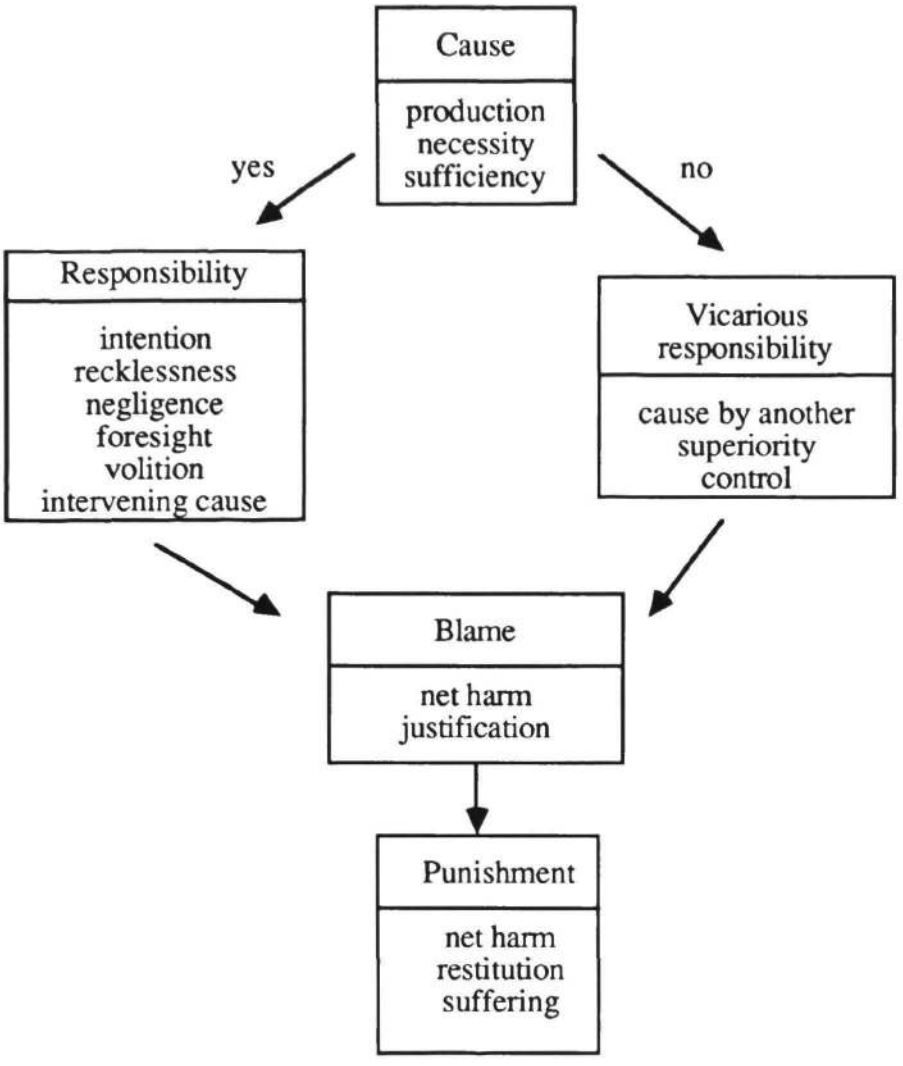


Figure 1. Major decisions and concepts in a theory of judging harm-doing

The first major judgment to be made concerns the causation of the harm. Causation is determined by a combination of generative and conditional information. On the generative view of causation, an effect is considered to be produced by some sort of transmission from the cause. Supplementing this approach is the *but for* (or sine qua non) test, which is widely used in jurisprudence. The *but for* test holds that a person's behavior is a cause of harm if and only if the harm would not have occurred without the person's behavior, thus focusing on the necessary conditions for harm. Some have argued for the use of sufficient conditions in judging causation: does the person's action distinguish the current harm-producing situation from some appropriate standard in which harm did not result?

The next major decision, responsibility, is determined by joint consideration of causation and excuses. One is held morally responsible for harm that he caused unless the harm was done accidentally (i.e., without intention, recklessness, or negligence), involuntarily (i.e., under external force), or without being able to foresee the resulting harm. Moreover, the causal chain leading from the action to the harm cannot be broken by some unforeseen event which exacerbated the harm.

The preferred way of judging intention is to match the accused's plan against the outcome. If his plan is known and the harm is included in the plan, then it is concluded that the harm was intended unless the harm was not caused as planned. If the actor's plan is unknown, then one falls back to objective heuristics such as valence, monitoring, and discounting. Did the harmful outcome have positive consequences for the actor, did he monitor his actions, or can intentionality be discounted because of alternate external causes? Recklessness is acting without due care coupled with high foreseeability of harm, unless the harm was intended. Negligence is acting without due care, but with a lower foreseeability of harm, unless the harm was intended or done recklessly.

Blame is a joint function of moral responsibility, the presence of some degree of net harm, and justification for the harm. If the accused is morally responsible for the harm and there is some net harm (i.e., more harm than benefit to the victim) then the accused is blameworthy, unless the harm can be justified. The distinction between moral responsibility and blame is somewhat subtle, but relies on the difference between excuses and justifications. Excuses, reflecting the concepts in the responsibility box in Figure 1, are offered when one admits to having caused harm, but does not accept responsibility for it. If such an excuse is accepted, then the issue of blame does not arise. Justifications come into play when one accepts responsibility for having caused harm, but denies that it was bad thing to have done, thereby avoiding blame, and perhaps earning credit.

In order to be justified, harm must achieve some goal, that goal must be more highly valued than not doing the harm, and the goal must be achievable in no less harmful way. These conditions are evaluated in sequence since a consideration of each one presupposes an appropriate decision on the preceding one. For example, if the harm achieves no goal, then there is no reason to consider whether any goal is more highly valued than not doing the harm.

The discussion has so far focussed on holding someone blameworthy for harm that he has directly caused. However, it is possible for blame to be assigned without direct causation. Such cases are typically understood as involving vicarious responsibility. A person is held vicariously responsible only when that person is in a superior position to the perpetrator or could have prevented the perpetrator from causing harm.

If the accused is blameworthy, then punishment can be assigned. Consistent with the retribution theory of punishment, punishment is directly proportional to the net amount of harm, scaled down by restitution the perpetrator has made, and the degree to which the perpetrator has suffered as a result of having caused the harm.

COMPUTATIONAL MODEL

A computer program was developed to simulate how the ordinary person (down to about 5-years-old) reasons about harm-doing. The program is called MR, for Moral Reasoner. MR provides a convenient way of rigorously specifying the psychological theory reviewed above and a technique for having the theory generate conclusions that can be compared with those produced by human subjects.

The version of MR used here is written in Lisp with rules implemented as boolean procedures, returning either *true* or *false*.[2] An English version of the rule dealing with moral responsibility is given as an example:

---

[2] A more conventional way to write this program would be as production rules in a production system interpreter. Several versions of MR have been done in just that way.

If & the accused produced the harm
    the accused's action was not accidental
    the accused's action was voluntary
    the harm was a foreseeable consequence of the accused's action
    there was no intervening cause of the harm
Then the accused is morally responsible for the harm
Else the accused is not morally responsible for the harm

The MR program accepts a case described in terms of categorical values on a number of features (e.g., foreseeability of the harm is high) and produces a series of conclusions on any of the other critical concepts in the model needed or requested.

EXAMPLE TRACE FOR A SINGLE CASE

The following illustrates how the MR program deals with a particular case. The case of Lynch vs. Fisher was tried in Louisiana in 1947 (Hart & Honoré, 1959):

A highway collision occurred through the negligence of accused, whereby a third party was trapped in his car and injured. The plaintiff, seeing the collision, went to help and finding a pistol on the floor, handed it to the injured man, who in a state of delirium through the shock of the accident, fired at plaintiff and wounded him.

A case is described to the MR program as a set of attribute-value pairs. The particular values for Lynch vs. Fisher[3] were:

((case-name lynch v fisher) (produce-harm ?) (necessary-for-harm y) (sufficient-for-harm n) (mental-state negligent) (careful n) (plan-known n) (plan-include-harm ?) (harm-caused-as-planned ?) (monitor y) (benefit-accused n) (foreseeability low) (external-cause n) (external-force n) (intervening-contribution y) (foresee-intervention n) (severity-harm 0.5) (benefit-victim 0) (achieve-goal n) (goal-outweigh-harm ?) (goal-achieveable-less-harmful ?) (restitution 0) (accused-suffer 0) (verdict g))

The output from the MR program is presented as a series of inferences. For Lynch vs. Fisher, the output was as follows: Case of Lynch v Fisher; accused caused the harm; no direct evidence that accused intended the harm; accused's intention to harm cannot be discounted; accused was not reckless; accused was negligent; no indirect evidence that accused intended the harm; accused did not intend the harm; the harm was not accidental; accused's action was voluntary; the harm was foreseeable; there was an intervening cause of the harm; accused is not morally responsible for the harm; accused is not blameworthy; disagree.

The accused caused the accident negligently, thus ruling out a pure accident. Also, the accused's actions were voluntary (i. e., not forced) and some kind of serious harm was

---

However, the Lisp version described here was found to be especially convenient for simulating large numbers of cases with damaged rules (see DAMAGING THE MODEL, below). The Lisp version of MR functions much like a backward chaining production rule interpreter.

[3] The symbols y, n, and ? refer to yes, no, and undecided, respectively. The symbol g refers to guilty, the verdict being used only to tabulate the rate of agreement between the program and the court.

foreseeable from negligent driving. However, because there was an intervening contribution to the harm, which was not foreseeable, MR decides that there was an intervening cause of the harm that mitigates responsibility and, thus, blame. In contrast, the court found the accused guilty.

EVALUATING THE COMPUTATIONAL MODEL

The model was tested on two large sets of actual cases of harm-doing. One set was based on legal cases in English and American law; the other set on cases recorded among traditional cultures that possess no codified legal system.

The first set consisted of the 95 most fully described legal cases in Hart and Honoré (1959), spanning the last five centuries of Anglo-American law. The overall proportion of agreement on MR's decision of blameworthy with a judicial decision of guilty was .84, $X^2 = 44.47, p < .001$.[4] This rate of agreement is gratifyingly high considering the difficulty of this set of cases, as indicated by the relatively low proportion of unsuccessful appeals, .61. The model agreed more with the final judicial decision than did the initial judicial decision, $X^2 = 12.15, p < .05$.

The second set contained 58 cases of harm-doing reported by anthropologists working in traditional, non-literate cultures around the world. The largest, single source was Pospisil (1958). The proportion of agreement was higher here (.97) than with the Hart and Honoré cases, $X^2 = 4.61, p < .05$, reflecting the fact that these cases were conceptually quite simple. This result suggests that the model implemented in MR does have some claim to cultural universality.

These above simulations, initially conducted blind with regard to the real life decisions, were useful in fine tuning the rule base, chiefly by repairing inconsistencies and anomalies in the rules.

DAMAGING THE MODEL

In order to determine whether each of the rules in the model is critical to matching real-life decisions, the cases were run again under conditions in which each rule was damaged. A rule was damaged by reversing the boolean value it returned: *true* if it was supposed to return *false*, *false* if it was supposed to return *true*. Only those rules leading up to the critical decision of blameworthy were damaged in this way.

The results for the 95 Hart and Honoré cases are presented as solid bars in Figure 2 in terms of proportion agreement with the final judicial decision. The proportion agreement produced by each type of damage was contrasted with that produced by the fully intact model using log-linear analysis. Damage to each rule did significantly lower the agreement rate (mean = .35, all $ps < .001$), except for those rules dealing with the mental state of the accused (mean = .83, all $ps > .5$). This revealed that for decisions of blame in cases of non-accidental harm (virtually all of the Hart and Honoré cases), it does not matter whether the accused acts intentionally, recklessly, or negligently. This seemed quite surprising until it was realized that these mental state concepts compensate for each other so that, even if one is damaged, another fills the gap.

---

[4] All of the statistics reported in this paper are based on log-linear analysis. Each result is reported as a 1 *df* Wald chi-square value.

Two rules are critical to understanding this compensation. One rule deals with accident, the other with intention:

If the protagonist's action was intentional, reckless, or negligent
Then the harm was not accidental
Else the harm was accidental

If there is direct or indirect evidence that the protagonist intended the harm
Then the harm was intended
Else the harm was not intended

As an example of compensation, even if the *intend* rule is damaged, the *reckless* or *negligent* rule may still prevent the harm from being viewed as accidental. As another example, even if the *strong-intend* rule (providing direct evidence of intention) is damaged, the *weak-intend* rule (providing indirect evidence of intention) may still lead to the conclusion that the harm was intended.

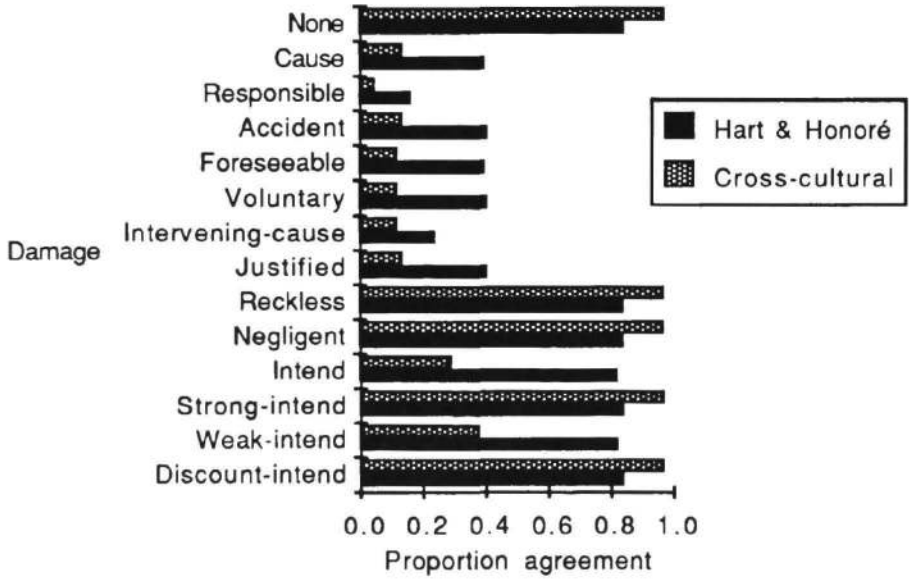


Figure 2. Proportion agreement with judicial decisions after rule damage.

The results for the 58 cross-cultural cases are likewise presented in hatched bars in Figure 2 in terms of proportion agreement with the actual decision. The proportion agreement produced by each type of damage was again contrasted with that produced by the fully intact model using log-linear analysis. The results were similar to those for the Hart and Honoré cases with two striking exceptions -- *intend* and *weak-intend*. The *intend* rule concludes that the harm was intended only if there is either strong or weak evidence of intention. The *weak-intend* rule concludes that there is weak evidence of intention if the accused's actions were neither reckless nor negligent, and one or more of the following is true: the accused's intending the harm cannot be discounted, the accused monitored his actions, or the harm benefitted the accused. A *strong-intend* rule concludes that there is strong evidence for intention if either the harm was described as intentional or the following all hold: the accused's plan was known, the accused's plan included the harm, and the harm was caused as planned.

Damage to the *intend* rule produced a proportion agreement of .82 in the Hart and Honoré cases, but only .29 for the cross-cultural cases, $X^2 = 37.32$, $p < .001$. Analogously, damage to the *weak-intend* rule produced a proportion agreement of .82 in the Hart and Honoré cases, but only .38 for the cross-cultural cases, $X^2 = 28.05$, $p < .001$. For the cross-cultural cases, damage to each rule did significantly lower the agreement rate as compared to the intact model (mean = .17, all $ps < .001$), except for the *reckless*, *negligent*, *strong-intend*, and *discount-intend* rules (mean = .97, all $ps > .5$).

This initially surprising discrepancy between the western and traditional cases can be explained by recalling that the latter cases are conceptually much simpler than the former. The Hart and Honoré cases were typically subtle and complex, turning on issues such as causation, intervening causation, negligence, or recklessness. In contrast, the cross-cultural cases were extremely straightforward, typically involving weak evidence for intention of the accused and not hinging on difficult issues such as causation, intervening causation, or alternative mental states such as negligence or recklessness. A typical cross-cultural case involved, for example, one person stealing another person's pig, and either making or not making restitution for it. Because there was no alternative mental state to compensate for damage to the *weak-intend* and *intend* rules in these latter cases, this type of rule-damage was fatal to proportion of agreement. In contrast, damage to the other mental state rules had little effect on proportion agreement since these other rules were rarely relevant to the cases.

CONCLUSIONS AND DISCUSSION

The simulations show that the MR program is computationally sufficient to qualitatively match human reasoning about harm-doing. They further suggest that the MR model and the psychological theory on which it is based have a large degree of cultural universality. Cultures undoubtedly vary in their value judgments about what constitutes what degree of harm and what sorts of justifications outweigh harms, but they do not appear to differ in the rules they apply to moral judgments about harm-doing. This universality could have important implications for explaining the development of these moral judgment rules.

The construction of the MR program was extremely useful in forcing a rigorous specification of an increasingly complex psychological theory. This was particularly true in terms of specifying how different parts of the theory ought to interact. Early simulations identified a number of inconsistencies and anomalies in the rule base. Corrections of these problems were then tested in subsequent simulations. It would have been extremely difficult to identify and test these issues without the benefit of a working computational model.

One unanticipated sort of interaction among of parts of the model was the degree to which mental state rules compensate for each other. Moreover, the extent of this compensation was found to interact with the subtlety of the case. For the relatively difficult cases found in books on western jurisprudence, the mutual compensation of mental state rules was extensive since many of the mental state rules were relevant to many of these cases. But for the relatively simple cases found in anthropological reports from traditional cultures, this mutual compensation disappeared since only a few of the mental state rules were relevant.

LIMITATIONS OF THE CURRENT MODEL AND SUGGESTIONS FOR FUTURE WORK

Although encouraging, these simulations do not constitute a very complete test of MR. They compare only one decision, blameworthiness, a decision that can be reached by a number of different paths through the rule base. More refined and more complete tests of

MR could be made by examining the intermediate decisions of ordinary subjects reading vignettes of legal cases.

Much more work is needed on the process of encoding the initial description of the case. Currently, the programmer translates the English version of the case into an attribute-value frame that the rules can use to make further inferences. It is likely, however, that subjects differ substantially in how they encode and interpret at least some cases. Such encoding differences would undoubtedly lead to differing conclusions about the case.

RELATED WORK

Pennington and Hastie (1988) investigated decision processes in simulations of trials by jury and constructed a theory of how jurors organize the information emerging during a trial. It is their view that the juror's main task is to construct a causal explanation of how the harm was produced and that the judge, through final instructions to the jury, often provides the responsibility and blame rules necessary to reach a decision on guilt or innocence. The emphasis of the present project is on explicating these rules as used implicitly by ordinary reasoners.

Thagard (1989) applied a connectionist model of explanatory coherence, ECHO, to two legal cases in which the prosecution and the defense advocated incompatible ways of explaining the evidence. Given a network of coherence and incoherence relations among propositions describing the evidence and competing claims of the case, ECHO propagated activation across the network to maximize the coherence of the network. ECHO focuses only on the issue of how the harm was caused.

Bain's JUDGE (in Riesbeck & Schank, 1989) is a case based reasoning program in the area of criminal sentencing. It uses rules to build up its case library. It has been our experience (via think aloud protocols and pointed questions) that, perhaps unlike judges, ordinary people have no extensive case libraries to draw on.

REFERENCES

Darley, J. M., & Shultz, T. R. (1990). Moral rules: Their content and acquisition. *Annual Reviews of Psychology*, **41**, 525-556.

Hart, H. L. A., & Honoré, A. M. (1959). *Causation in the law*. Oxford: Clarendon Press.

Pennington, N., & Hastie, R. (1988). Explanation-based decision making: Effects of memory structure on judgment. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, **14**, 521-533.

Pospisil, L. (1958). *Kapauku Papuans and their law*. New Haven: Yale University Press.

Riesbeck, C. K., & Schank, R. C. (1989). *Inside case-based reasoning*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Shultz, T. R., & Schleifer, M. (1983). Towards a refinement of attribution concepts. In J. Jaspars, F.D. Fincham, & M. Hewstone (Eds.), *Attribution theory and research: Conceptual, developmental and social dimensions* (pp. 37-62). London: Academic Press.

Thagard, P. (1989). Explanatory coherence. *Behavioral and Brain Sciences*, **12**, 435-467.

The Mechanism of Restructuring in Geometry

Stellan Ohlsson

Learning Research and Development Center
University of Pittsburgh, Pittsburgh, PA 15260
stellan@vms.cis.pitt.edu

Abstract. Restructuring consists of a change in the representation of the current search state, a process which breaks an impasse during problem solving by opening up new search paths. A corpus of 52 think-aloud protocols from the domain of geometry was scanned for evidence of restructuring. The data suggest that restructuring is accomplished by re-parsing the geometric diagram.

Introduction. A wide variety of problem solving processes have been analyzed in terms of heuristic search (Newell & Simon, 1972). For example, in geometry proofs the geometric theorems (operators) are applied to the mental representation of the diagram (the knowledge state) until the desired proposition (the goal state) has been attained (Anderson, 1981). The stepwise character of heuristic search contrasts with the Gestalt hypothesis that problem solving proceeds through (a) an initial, unsuccessful, attack on the problem, (b) a more or less protracted impasse, and (c) a restructuring of the problem, which is typically, but not necessarily, followed by insight (Ohlsson, 1984a).

Several attempts have been made to reconcile the information processing and Gestalt hypotheses. Simon (1966) proposed that it helps to sleep on a problem, because goal tree information is forgotten faster than problem information. After a pause, a new goal tree is built on the basis of more knowledge about the problem. Langley and Jones (1988) interpret an impasse as a failure to retrieve the relevant problem solving operator. Insight occurs when some *external* stimulus causes enough activation to spread to that operator to allow its retrieval. A related hypothesis claims that insight occurs when an appropriate analogy is retrieved (Keane, 1988). Both the differential rate of forgetting hypothesis and the spread of activation hypothesis require that the problem solver moves attention away from the problem, and so cannot explain insight during *ongoing* problem solving. Several researchers have proposed that problem representations can be improved by the construction of macro-operators (Amarel, 1968; Korf, 1985). Koedinger and Anderson (in press) have proposed the related idea that geometry experts combine geometric theorems into larger inference schemas, called *diagram configuration schemas*, which allow them to find a proof without step-by-step search of the proof space. The macro-operator and diagram configuration hypotheses explain expert performance, but they do not explain insights by novices. All of these hypotheses locate restructuring in the *processes* of problem solving.

In contrast, I have proposed that restructuring involves a change in *the mental representation of the current search state* (Ohlsson, 1984b). A change in the

representation implies that objects, relations, and properties which initially are seen as instances of certain concepts are being re-encoded as instances of other concepts. For example, an object which is initially encoded as a *hammer* might in the course of problem solving become re-encoded as a *pendulum weight*, a *line* may be re-encoded as a *triangle side*, and so on. Re-encoding a search state changes the set of operators which are applicable in that state, and thus breaks an impasse by opening up new search paths. A similar theory has been proposed by Kaplan and Simon (in press) to explain restructuring in the Mutilated Checker Board Problem. The critique by Montgomery (1988) does not touch those aspects of the theory that are of main concern in this paper. The purpose of the present paper is to provide evidence for re-encoding from the domain of geometry, and to propose a mechanism for re-encoding in that domain.

Table 1. Geometric theorems acquired by the subjects.

Theorem 1. Supplementary angles are congruent.

Theorem 2. Vertical angles are congruent.

Theorem 3. The supplementary angle of a right angle is a right angle.

Theorem 4. If two angles and their common side in one triangle are congruent to the corresponding angles and their common side in another triangle, then the two triangles are congruent.

Theorem 5. If two sides in a triangle are congruent, then their opposite angles are congruent; and vice versa.

Method. Three undergraduate psychology students participated in an experimental course in elementary geometry. The experimenter saw each subject individually in sessions that lasted approximately one hour each. The subjects learned basic theorems of plane geometry, the first five of which are shown in Table 1. A typical session began with free recall of previously learned theorems, continued with the introduction of new theorems, and ended with problem solving practice. The subjects had the theorems available during problem solving, and they were instructed to think aloud. The data consist of 52 think-aloud protocols, representing a total of approximately nine hours of problem solving effort. The protocols were scanned for the occurrence of restructuring events. Ten such events were found. The three most informative events will be analyzed below. They illustrate deliberate restructuring, goal driven restructuring, and restructuring in response to a hint.

Case 1: Deliberate restructuring. Subject S3 was given the problem in Figure 1 after she had studied Theorems 1-5 (see Table 1). She began by proving that triangles

AED and BEC are congruent, and then entered an impasse. In fragments F65-F67 (see Table 2) she deliberately sets out to *see* the problem from *many viewpoints*. The process of restructuring proceeds through three steps. First, she mentally cuts the figure along the diagonal CA, forming the triangles CDA and CBA (F68-F70). She then mentally cuts the figure along the other diagonal, forming the triangles DCB and DBA (F71-F74). Finally, she keeps one triangle from each pair, as it were, and sets herself the task of proving them congruent (F75-F77). Figure 2 gives a diagrammatic analysis of the process. The geometric objects perceived by the subject are drawn in bold lines, while the rest of the diagram is drawn in broken lines. Restructuring was not followed by insight in this case. The subject worked on the problem for twelve minutes without solving it.

Table 2. Protocol excerpt from Subject S3.

F65. but perhaps one can see this in some other way also
F66. one can perhaps see this from many viewpoints here
F67. now we shall see
F68. one can see it as
F69. CDA and CBA
F70. triangles
F71. one can see it on
F72. DCB and DBA instead
F73. yes exactly yes
F74. those two
F75. well
F76. now I can see this in another way
F77. CDB and CAD ought to be congruent here in some way

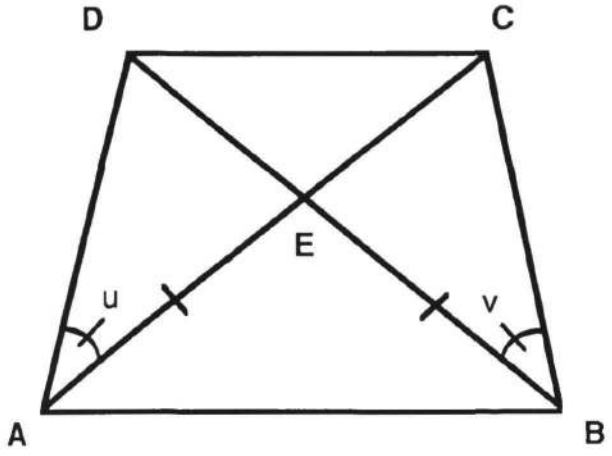
Case 2: Goal-driven restructuring. S1 was given the problem in Figure 1 as his first problem after studying Theorems 1-5 (see Table 1). S1 misunderstood the goal of the problem to be to prove that angle ADC is congruent to angle BCD. When the protocol excerpt in Table 3 begins, he has proved that angles EDA and ECB are congruent by proving them corresponding parts of the congruent triangles EDA and ECB. He then sets himself the goal of proving that the *remaining parts*, i. e., angles EDC and ECD, are equal (F43). His plan is to prove that they are equal by proving that the *sides* of the triangle EDC are equal (F42-F45). This goal is reformulated as proving that the triangle EDC is *isosceles* (F46-F47). This view of the problem leads to an impasse (F48-F49). Prompted by the experimenter to continue to think-aloud, he states that he is thinking about *the same problem* but from *another angle* (F50-F52): he has re-encoded ED and EC as *lines* (F54). The goal is still to prove them congruent (F53-F55). He suddenly realizes that ED and EC are corresponding *sides* of

the two triangles EDA and ECB, which he has already proved congruent (F56-F61). Figure 3 shows a diagrammatic analysis of the process with perceived geometric objects in bold lines and the rest of the diagram--the background--in broken lines. The subject quickly completed the correct solution.

Table 3. Protocol excerpt from Subject S1.

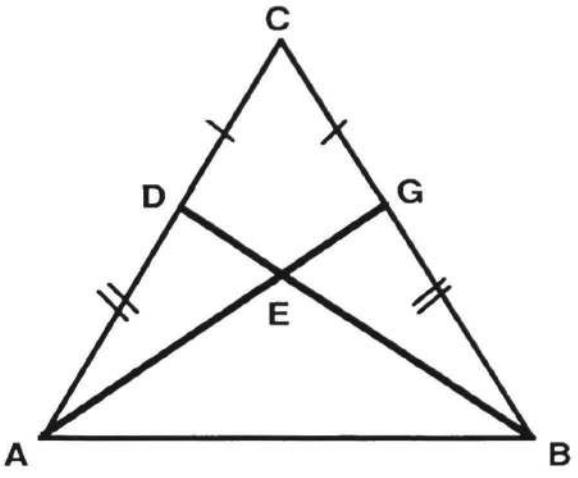
F42. yes now I am thinking about whether one can prove that these two sides [DE, EC] are equally long
F43. because if they are then those two angles [EDC, ECD] which are just the remaining parts of those angles which I want to get [ADE, BCD] must be equally long
F44. so then this and that angle [ADE, BCD] must be equally big
F45. and then the problem is solved
F46. so it is now a question of proving that it is isosceles
F47. that triangle [EDC]
F48. and that I cannot
F49. but perhaps one can do it in some other way
(What are you thinking?)
F50. well now I am thinking
F51. well it is the same problem
F52. but from another angle
F53. yes if this one
F54. is those two lines [ED, EC] are equally long
F55. I am thinking
F56. yes but they must be
F57. since they are parts of
F58. it is congruent
F59. these two here are congruent [triangles EDA, ECB]
F60. and it is [ED, EC] corresponding sides in the triangles [EDA, ECB]
F61. therefore these two sides [ED, EC] are equally long

Case 3: Hint-driven restructuring. S2 attempted Problem 2 (see Figure 4) after having learned the five theorems in Table 1, plus four others. She decided to prove triangles AED and BEG congruent and quickly reached an impasse. The protocol excerpt in Table 4 begins when the experimenter gives her the hint that there are other pairs of triangles in the figure that might be congruent. She first rejects this suggestion (F113-F115). She then runs through the triangles in the figure (F113-F121), and concludes that there are no other congruent triangles in the figure (F121). She then suddenly sees the triangles AEG and BDA (F123-F124). Figure 5 shows a diagrammatic analysis of the process with perceived geometric objects drawn in bold lines and the rest of the diagram drawn in broken lines. In spite of this restructuring, the subject failed to solve the problem.



Prove angles ECD and CDE congruent.

Figure 1. Problem 1.



Prove line segments AG and BD congruent.

Figure 4. Problem 2.

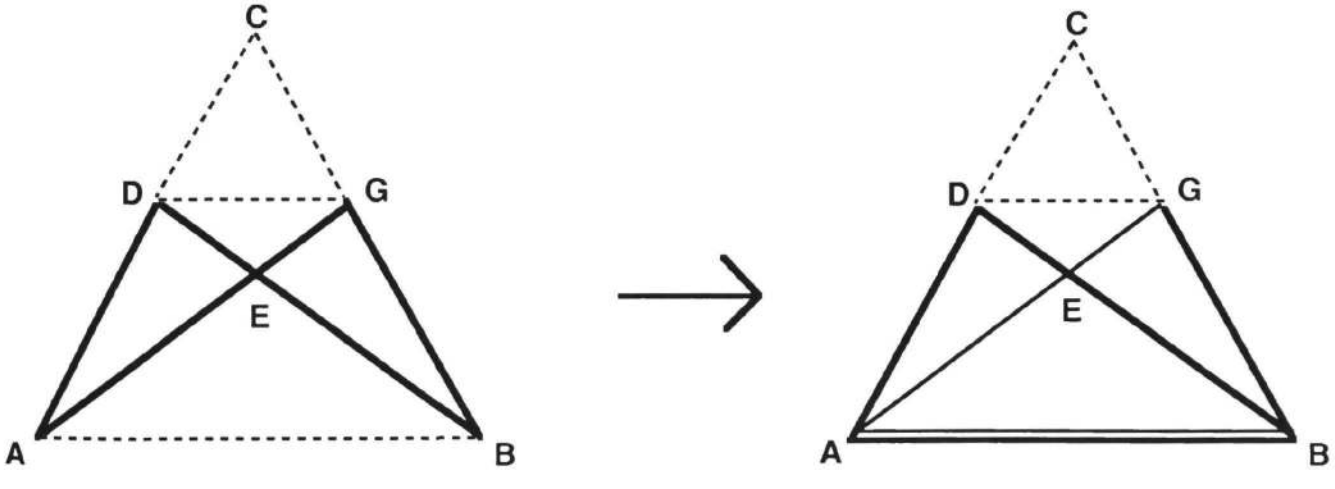


Figure 5. Analysis of S2's re-encoding process. Perceived geometric figures are drawn in bold lines, the rest of the figures in broken lines.

Table 4. Protocol excerpt from S2.

(What other triangles could be congruent?)
F109. what others
F110 could there be others which are congruent
F111. huh
(That could be. You have now been working the hypothesis that the whole point is to prove that those two triangles [AED, BEG] are congruent.)
F112. yes
(And just now you reached the conclusion that you cannot do that with the information you have. Can you find two other triangles which one can find which one could believe could be congruent?)
F113. congruent exactly alike
F114. no that is impossible there are no others
F115. it cannot be
F116. there are only one other
F117. also hypothetically then this line here
F118. then there are two here
F119. and those two here can surely never be congruent
F120. these two here can surely never be congruent
F121. no I do not understand that
F122. but
F123. now I see it
F124. I have forgotten this one here [AGB or BDA]

Discussion. The restructuring process revealed in these three protocol excerpts consists in *re-encoding the given figure*. The diagram--the set of lines on the paper--contains within it a large number of different geometric objects (angles, sides, triangles, etc.). Only some of those geometric objects are perceived at any one time. The others recede into the background. In particular, if a line configuration is perceived in one way, then alternative encodings of that same line configuration recede into the background. Restructuring consists of switching to one of the alternative encodings. How does the switching mechanism work? The data suggest that re-encoding is done by *re-parsing* the diagram. During initial problem perception complex objects (e. g., triangles) are constructed out of simpler objects (e. g., lines). This process is a search through a *description space* (Ohlsson, 1984b). Alternative interpretations of the perceptual information are possible, so some choices are made, resulting in a particular encoding of the given diagram. When an impasse forces the problem solver to re-encode the problem, he/she backs up in the description space, dismantles his/her previous encoding, and traverses another path through the description space. This process breaks an impasse by allowing other operators (geometric theorems) to apply to the current state. Restructuring is a rare

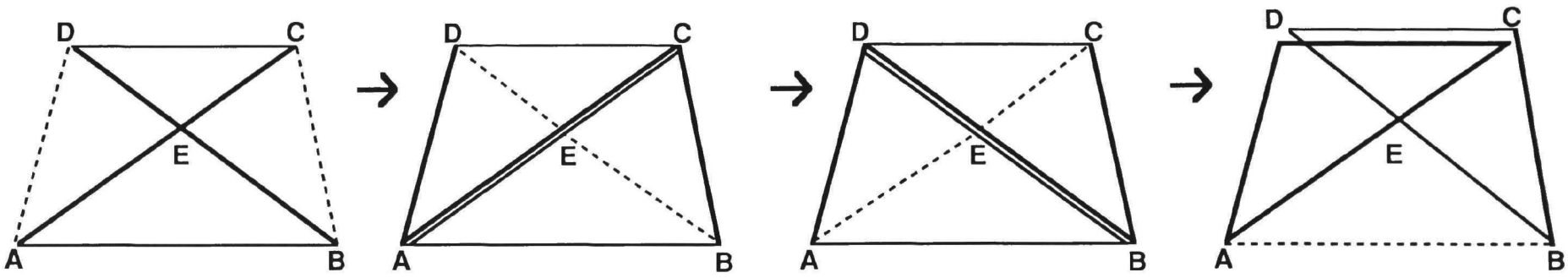


Figure 2. Analysis of S3's re-encoding process. Perceived geometric objects are drawn in bold lines, the rest of the figures in broken lines.

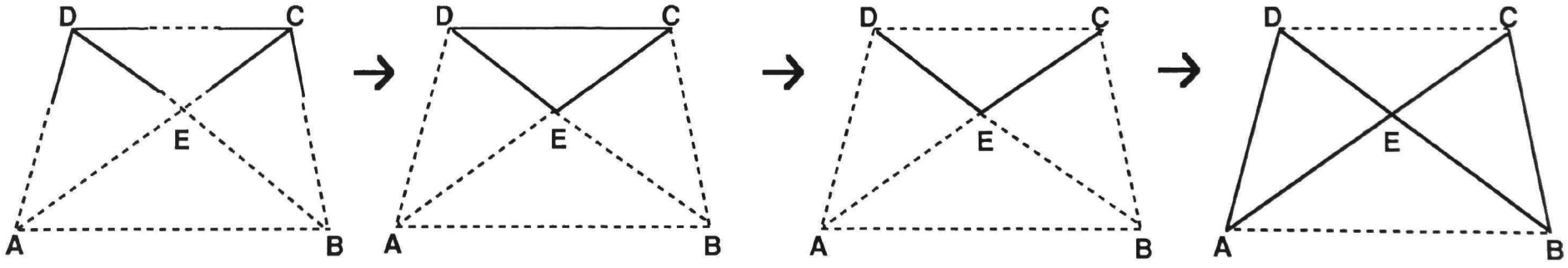


Figure 3. Analysis of S1's re-encoding process. Perceived geometric figures are drawn in bold lines, the rest of the figures in broken lines.

event: There was approximately one restructuring event per hour of problem solving effort in the present study. Restructuring does not necessarily lead to insight: In two of the three excerpts presented above, the subject failed to solve the problem. This study supports the idea that diagram parsing is central in geometry (Koedinger & Anderson, in press), but the validity of the re-parsing mechanism for other domains than geometry remains an open question. For example, a different mechanism seems to be responsible for re-encoding of the Mutilated Checker Board Problem (Kaplan & Simon, in press).

Acknowledgement. Preparation of this paper was supported, in part, by ONR grant N00014-89-J-1681. No endorsement should be inferred.

References

Amarel, S. (1968). On representations of problems of reasoning about actions. In D. Michie (Ed.), *Machine intelligence. Vol 3.* (pp. 131-171). Edinburgh, UK: Edinburgh University Press.

Anderson, J. R. (1981). Tuning of search of the problem space for geometry proofs. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence* (pp. 165-170). Vancouver, Canada: University of British Columbia.

Kaplan, C. A. & Simon, H. A. (in press). In search of insight. *Cognitive Psychology.*

Keane, M. (1988). *Analogical problem solving.* New York, NY: Wiley.

Koedinger, K. R. & Anderson, J. R. (in press). Abstract planning and perceptual chunks: Elements of expertise in geometry. *Cognitive Science.*

Korf, R. E. (1985). *Learning to solve problems by searching for macro-operators.* Marshfield, MA: Pittman.

Langley, P. & Jones, R. (1988). A computational model of scientific insight. In R. J. Sternberg (Ed.), *The nature of creativity. Contemporary psychological perspectives* (pp. 177-201). Cambridge, MA: Cambridge University Press.

Montgomery, H. (1988). Mental models and problem solving: Three challenges to a theory of restructuring and insight. *Scandinavian Journal of Psychology, 29,* 85-94.

Newell, A. & Simon, H. A. (1972). *Human problem solving.* Englewood Cliffs, NJ: Prentice-Hall.

Ohlsson, S. (1984a). Restructuring revisisted. I. Summary and critique of the Gestalt theory of problem solving. *Scandinavian Journal of Psychology, 25,* 65-78.

Ohlsson, S. (1984b). Restructuring revisited. II. An information processing theory of restructuring and insight. *Scandinavian Journal of Psychology, 25,* 117-129.

Simon, H. A. (1966). Scientific discovery and the psychology of problem solving. In *Mind and cosmos: Essays in contemporary science and philosophy. Vol. III.* (pp. 22-40). Pittsburgh, PA: University of Pittsburgh.

Noticing Opportunities in a Rich Environment

Matthew Brand and Lawrence Birnbaum

Northwestern University
The Institute for the Learning Sciences and
Department of Electrical Engineering and Computer Science
Evanston, Illinois

Abstract

Opportunistic planning requires a talent for noticing plans' conditions of applicability in the world. In a reasonably complex environment, there is a great proliferation of features, and their relations to useful plans are very intricate. Thus, "noticing" is a very complicated affair. To compound difficulties, the need to efficiently perceive conditions of applicability is simultaneously true for the thousands of possible plans an agent might use. We examine the implications of this problem for memory and planning behavior, and present an architecture developed to address it. Tools from signal detection theory and numerical optimization provide the model with a form of learning.

1 The Problem

An agent operating in a rich and rapidly changing environment must constantly monitor the environment as it acts, and be prepared to deal with opportunities and obstacles as they come up. The need to interact with rather than simply act upon the world presents a large variety of constraints on the computational behavior of a planner. We have tried to use these as guidelines for the design of an architecture for the pusuit of multiple goals and plans in a realtime environment.

A convenient route to the issues involved in realtime multiplanning lies in the colloquial language of opportunism. Our daily life is filled with myriad chances to satisfy or advance our goals. How we fare in the world has much to do with our ability to "notice" opportunities, "seize" them, and make use of them before the "window" of opportunity closes. Each of these idioms points to some of the many constraints that bear upon the design of a realtime multiplanner.

In particular, "noticing" means recognizing useful resources in the complexes of low-level, noisy sensory features available to the planner. Not all resources are worth attention; only those that enable plans which serve active goals. Thus, although "noticing" appears to be a perceptual process, its operation depends on decisions that must be informed by the high level goals of the agent. "Seizing" refers to the overhead of selecting a plan and executing it: choosing amongst competing plans and goals, verifying the opportunity, and assigning additional resources to the plan execution. "Window of opportunity" refers to the transience of opportunities: they have to be discovered and acted upon as suddenly as they appear and the agent may have to deal with their disappearing just as suddenly.

In short, realtime multiplanning presents three general constraints:

- The agent must incorporate an opportunity oriented planner, rather than a one-shot or agenda-oriented planners. However, being opportunity-driven does not mean that its behavior is determined bottom-up from perception: actions and indeed perceptions need to be goal-motivated. This is the *integration* constraint: The planner needs to adjudicate between bottom-up opportunities and top-down desires and expectations (see, e.g. [Birnbaum 86]).
- The agent must perform a quick and efficient mapping from surface features to applicable plans. This is the *time* constraint: The agent

must have a fast and fairly constant reaction time in any situation.

- The agent must be constantly replanning as external conditions and internal goals change. This is the *flexibility* constraint: The agent must be prepared at any time to execute, suspend, resume, or abandon a plan in the face of novel circumstances.

These constraints have guide the design decisions that form the system described below.

2 Real-World Constraints

Most compelling in the design of a realtime planner are issues of computational efficiency. The environment will tolerate only a narrow range of reaction times on the part of the agent, and the shorter the better. The agent must move from low-level perceptual features to high-level decision processes in as few steps as possible. Traditional planning tools, especially those based on search, are undesirable because they require indeterminate and exponential time to find or construct applicable plans. An animal does not stop to think how it can relate the sighting of prey to the get-nourishment goal. An efficient alternative to search is matching against rich libraries of planning information. Making this work, however, necessitates unusual commitments vis-à-vis memory, processing, and architecture.

2.1 Memory Issues

In order to be able to recognize which circumstances constitute an opportunity for the inception or resumption of a plan, that plan must already be available–and quite some detail–in the agent's memory. At the very least, it must be present in enough detail for its conditions of applicability to be quickly consulted and compared to the world. This is especially important because of the large number of potentially useful plans. The typical range of activities in an agent's "everyday" behavior will likely require hundreds or thousands of precompiled plans. We call the need to have plans easily consulted the *accessibility* constraint.

Given that we want plans to be tightly coordinated with the perceptual apparatus that cues them, it will prove useful to think of a plan coupled with the computing elements that recognize its conditions of applicability as a unit: a *behavior*. When we talk of a planner endowed with sensors and effectors, we talk of a collection of behaviors which we want to combine in ways salutary to the planner's goals. As with [Agre 88, Agre & Chapman 87, Firby 89, Maes 89] we are interested in the consequences of reinterpreting the planning task as a matter of *coordination of behaviors* rather then the *synthesis of plans*.

How specific should behaviors be? In any complex environment, behaviors do not have unique conditions of applicability; there is always some generality which cannot be captured in simple lists of features. An animal in a forest need not have one behavior for picking up edibles and another for picking up nesting materials. Having both in memory simply adds computational cost to the task of plan selection, probably more than is saved by simple feature matching. The plans in the agent's repertoire should be flexible, even though this makes recognizing each plan's conditions of applicability in the environment more expensive. There is a tradeoff between ease of opportunity recognition and generality of behaviors.

Schemes for flexible plans come with a number of extra burdens: plan synthesis requires search, plan modification requires extensive knowledge about planning, and abstract plan roles require complicated type checking. All three, however, violate the accessibility constraint. Consequently the planner's memory is strongly biased for ease of matching and against "abstract" or "universal" planning methods.

2.2 Attention Issues

Plans exist for many different time scales. Few are executed *en tous* and without interruption. Many cover spans of time so large that an agent cannot afford to execute them without interruption. For example, one cannot follow the house-building plan without breaks to entertain the eat-and-be-refreshed plan. Some plans are so long-term, such as write-thesis, or of such low priority, such as pick-up-money-from-sidewalk, that more time passes during interruptions than during execution.

Proverbially, we want an agent to be able to "walk and chew gum at the same time." This means dividing attention between many plans that are concurrently being executed, some in parallel, some in dovetailed sequence.

The idea of precompiled plans is helpful here too. If a plan may not be executed *en tous*, then conditions of applicability need be computed not only for its beginning, but at any point where it may be interrupted and later resumed. The more concrete

the description of the plan, the less costly this calculation is.

The possibility of interruption also argues for plans that are as short as possible. Rather than try to execute a full plan for building a shelter, a agent will find it easier to have a building plan broken down into its component behaviors, which it can splice together as their conditions of applicability become true in the environment. This has a number of advantages for the planner. Breaking a plan into fragments provides convenient interruption points with precomputed requirements for resumption. Plan fragments can be recombined and reordered according to the vagaries of environmental change.[1]

3 Architecture

In perceiving the world, the agent should calculate an optimal set of features for its purposes–meaning it should compute as few features as possible to identify applicable plans and choose between them. Features should only be computed when necessary *vis-à-vis* likely plans, and when necessary to many plans, a feature should only be computed once.

An efficient architecture which addresses the two priorities of shared computation of features and arbitrary mappings is a feed-forward network. Low-level features pass their information up to increasingly complex features which in turn cue and recue candidate plans. This opens the door to parallelism. However, we cannot assume that the "magic" of massive parallelization will somehow make this network tractable. Limited resources mean that a limited number of features can be computed at any given time.

Optimization in this context means making choices about which nodes in the network will receive computational resources, and which will have to remain dormant. Dormancy means that the feature retains its old value, and is increasingly likely to become incorrect as the environment changes. We wish to construct a description of how efficiently the network is using its features (and thus retrieving appropriate plans). Any network control strategy that optimizes that function will thus provide a means for intelligent attention focusing.

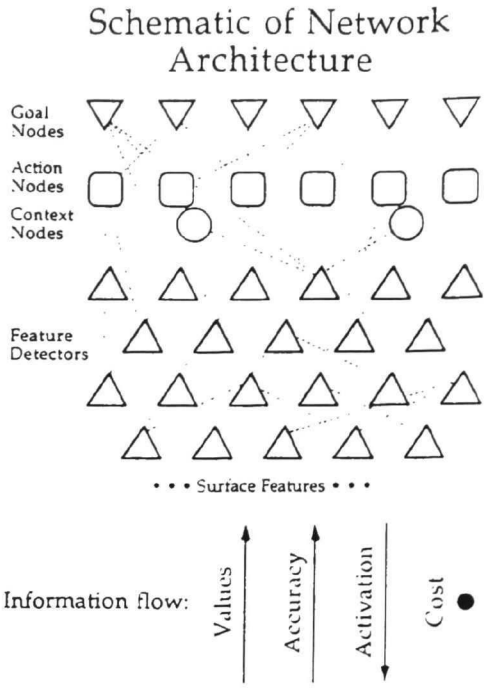


Figure 1: Connectivity and information flow in the planning network.

We have arranged the knowledge sources of our planner in a network for this purpose (see figure 1). Each node represents a kind of computation the planner can do: either computing a feature of executing a plan step. The links between nodes are pathways through which nodes interact, passing information and competing and cooperating to control the effectors and computational resources of the agent.

4 Representation

There are four basic types of knowledge in the agent: goals, action sequences, feature detectors, and memory pools. Between them pass three kinds of information: values, accuracy estimates, and activation.

4.1 Information Flow in the Planner

Values are scalars that are passed up from lower-level nodes to higher-level nodes. They indicate whether the feature computed by some node was present in the world last time the node was computed.

Accuracy estimates are scalars indicating how likely it is that a feature has been computed using

[1] This also relieves the agent of much of the memory burden involved in remembering its place in a long process. The environment is exploited as a mnemonic device, like a mechanic layings out pieces from a machine to mark his plac in the disassembly and reassembly of a machine.

obsolete information. Features can be inaccurate because subfeatures they depend upon may have been dormant while something in the world has changed. Accuracy estimates flow up the network along with values, and are reduced as they pass through each obsolete node.

Activation, also a scalar, flows down through the network over weighted connections. It is used to determine what parts of the network should be computed. Essentially, the activation at a node is an indicator of how interested the planner is in committing computational resources to it.

4.2 Knowledge Structures in the Planner

At the top of the network, goals provide the impetus behind the agent's choices. They are not predicates describing world states, as is the tradition in planning. Rather they are generators that supply activity to the action sequences which satisfy them and the features which recognize conditions in which they are satisfied.

Feature detectors are arbitrary functions which read in sensory data and/or other feature values and output a value indicating some state of affairs in the environment and the planner.

Action nodes encapsulate planning knowledge. Moving upwards, they signal their execution status to the goals that they serve, inhibiting or satisfying them. Moving downwards, they pass activation on to the feature detectors that correspond to their preconditions, priming them. From goal nodes they receive activation, and from feature detectors they receive information about their applicability. As the agent monitors the world, each action node jockeys for the right to determine the next action.

There is no basic unit of plan representation inside an action node node. Instead, these nodes hold plan fragments, single operators, and occasionally even entire canned plans.

Plans and plan fragments are expressed wholly in terms of effector instructions. There is no explicit representation of subgoals, else the planner would have to resort to search. Although this would seem to severely limit the potential sophistication of the plans the agent can express, we are take hope from two hypotheses. The first is that a large range of interesting and useful behaviors *can* be achieved under this limitation–perhaps even enough for a reasonable simulation of animal behavior.

The second is that where subgoaling really is necessary, we may expect subgoaling-like behavior in the way the agent sequences its behaviors. The reason why is that, should action node A need a precondition achieved, it will prime the feature detectors that are looking for satisfying conditions in the world. If action node B has those feature detectors as a description of what it achieves, and they are partially true, node B is likely to be executed opportunistically. Once this has happened, node A's preconditions have been satisfied, and it will be executed. Though not dependable, this scenario points a way to rudimentary subgoaling.

Memory pools are frozen contexts. Each name a plan in execution, the goal it serves, and the resources that are tied to the various roles in the plan. They point to the feature detectors that recognize their role fillers. In a carpenter's attach plan, for example, roles might be Object1, Object2, SupportingSurface, MeansOfAttachment, ClampingDevice, and Tool. Their fillers could be, respectively, a broken chair, its leg, a clear space of floor, wood glue, rope, and the right hand.

Memory pools are used to ensure that a plan can be continued. They are quite expensive, and tend to evaporate after time. This is because an agent must have a means of forgetting thwarted plans after some passage of time, in order to be reasonably free to exploit new opportunities.

5 Process Model

The steps of a typical planning cycle are:

- Choose the features to be recomputed.
- Recompute features.
- Propogate information throught the network.
- Choose an action node to control effectors.
- Execute one instruction from the node's plan fragment.
- Construction, update, or activate a memory pool to hold the current context.

The important steps in this cycle are the selection steps. These choices govern the agent's external and internal attention, and are governed by the parameters which flow through the network.

In addition to the three parameters described above–value, activation and accuracy–each node has a stationary parameter: cost. It represents the computation expense associated with that node, and is used to determine when the feature is worth recomputing. Low-level features, which are worth comput-

ing often because they are close to sensory information and are the basis for all other computation, are assigned low costs. High-level features, which are in a sense more speculative, are assigned high costs. In this way the network is biased to pay close attention to the environment.

How is the distribution of computation question decided? In the case of feature detector nodes, the feature is recomputed if

$$f(n) = \frac{activation_n - cost_n}{1 - accuracy_n} > T_F$$

where T_F is a global threshhold.[2] This equation expresses the main economy of feature computation. It simply ensures that the features are recomputed when they receive large amounts of activation or are very likely to be obsolete. A similar function $F(n)$, governs action nodes and memory nodes.

The purpose behind these parameters, besides providing the basis for interaction between knowledge sources, is to allow the planner to judge what is worth computing. The basic premise is that some kinds of computation are more expensive than others. Passing the parameters around the network and calculating $f(n)$ and $F(n)$ are cheap; we pretend that they are properties of some imaginary computational substrate. More importantly, they happen in constant time, regardless of the agent's circumstances. Actually computing the features also takes constant time, because a fixed fraction of the network is considered.

Computing the features is held to be more expensive than propagating information, and thus we limit the number that are computed each cycle. This is not just to conserve computation: if too many feature nodes are active, too many action nodes that will merit consideration. Thus as features receive inadequate activation they report increasingly obsolete information, and decay to an off-state, in which they are incapable of supporting any action node's bid for execution.

An action node is queued for evaluation if $F(n) = f(n) + K \cdot s(n) > T_A$, where K is a global coefficient,[3] $s(n)$ is a weighted vote of precondition features for that node, and T_A is another global threshhold like T_F. The node with greatest $F(n)$ is picked out of the queue, its role bindings verified, and the first unexecuted operator in it is used to control the effectors. If the role fillers fail to verify, that node can be "suspended" by construction of a memory pool, or a different node can be taken off the queue. Only a fixed number of nodes are checked, and the queue is discarded after each cycle.

Action selection is also constant in time relative to the input, and is logarithmic in the size of the network. Cycle time is thus constant, and adjustable via the two global threshholds T_F and T_A.

When the current action node is usurped by a node with greater $F(n)$, it is suspended. A memory pool is set up for it which saves the context at time of suspension. This memory pool is treated like an action node in subsequent cycles: it receives activation from the action node's sponsor goal, it is queued and evaluated like action nodes, and when it is selected the action node it is attached to resumes execution. Memory pools have a decay term in their $F(n)$ which grows increasingly negative with time, making their resumption less and less likely. When their $F(n)$ itself becomes negative, they "evaporate" and the planner loses any trace of having been in the midst of their plans.

6 Learning

Optimization is plausible in our architecture because the flow of information is restricted to specific pathways between knowledge structures, greatly simplifying credit assignment, and because information is restricted to scalars, allowing us to construct a mathematical characterization of the network's performance.

In the network, goals are served both by feature detectors, which report goal satisfaction and action nodes, which attempt goal satisfaction. Using the feature detectors as reference signals, it is easy to collect statistics on an action node's success *vis-à-vis* its sponsoring goal. The first optimization open to us is to strengthen the weights on activation links between goals nodes and the action nodes that most reliably serve them. This allows local improvements in the ability to select plans.

A more subtle and important optimization applies to the relationship between action nodes and

[2] T_F controls what fraction of the network's features are recomputed in any given cycle. A higher threshhold makes the agent like a panicked animal; it needs to respond very quickly and thus can only attend to its most immediate goals and sensations. A lower threshhold makes the agent slower, but more alert to uncommon opportunities.

[3] K allows us to balance the relative importance of need, expressed by activation, and opportunity, expressed by sensory information.

the feature nodes that detect their conditions of applicability. Here it is important to learn which features best cue an action node, and thus are most deserving of the activation that the action node can distribute. Here we use some simple tools from signal detection theory [Tanner, Swets, & Green 56, Green & Swets 66] to provide a platform from which we can do numerical optimization.

By comparing records of feature node firings with the statistics described above, we can tabulate the feature's utility (via the plan fragment) to the goal in terms of hits $p(F \cdot A)$, false alarms $p(\neg F \cdot A)$, misses $p(F \cdot \neg A)$, and correct rejections $p(\neg F \cdot \neg A)$. F refers the the presence of the feature and A refers to the decision to take an action on the basis of that feature.

From the costs and activation assigned to each node we can construct a payoff matrix. Using these and the statistics described above, we can adapt from signal detection theory the equation describing the expected value of relying upon that feature:

$$EV = W_{A\rightarrow F}p(F\ \ A) + C_A p(F \cdot \neg A) + C_F p(\neg F \cdot A) + C_G p(\neg F\ \ \neg A) \quad (1)$$

In this notation, $W_{A\rightarrow F}$ is the weight on the activation link from the action node to the feature node, C_A is the fixed cost of evaluating the action node, C_F is the fixed cost of the feature node, and C_G is a global cost assessed for missing opportunities.

By toggling $W_{A\rightarrow F}$ between two close values and collecting statistics on the relative values of EV, we discover local information about the first derivative of the actual probability curve relating feature firing to plan appropriateness.[4] For local hillclimbing, we can simply choose to remain with the value of $W_{A\rightarrow F}$ that produces the greater EV

More interesting is the case when we apply a global optimization technique. We are investigating the use of simulated annealing [Kirkpatrick, Gelatt & Vecchi 83] in the selection of weights. Under simulated annealing, one chooses one value of $W'_{A\rightarrow F}$ over another $W_{A\rightarrow F}$ with the probability

$$e^{-\frac{EV'-EV}{T}}$$

for some temperature T which determines the "volatility" of the decision. This is a particularly interesting prospect because we may be able to regulate the temperature of the annealing decisions as a function of the accuracy reported at the nodes involved.

[4] There is a strong assumption of Gaussian noise in the functioning of the network in order for this to work. We can rely on the obsolescence of some features for noise, but it is not clear that it will approximate a Gaussian.

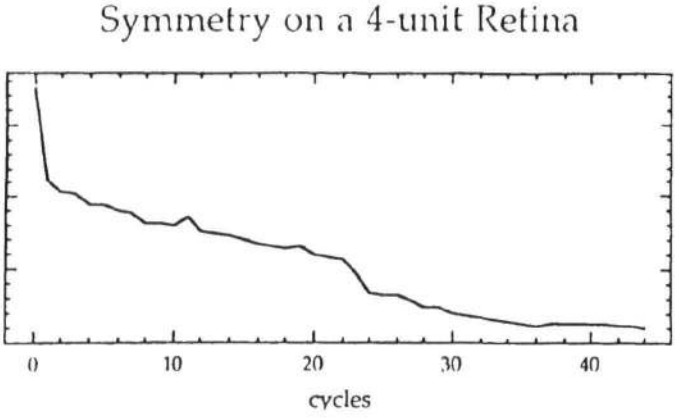


Figure 2: Error curve of a primitive variant of the optimization algorithm applied to learning symmetry in a 4-bit vector.

Simulated annealing is appropriate here because it allows us to globally optimize the interactions between "opaque" functions. By "opaque" we mean that the functions in the nodes are arbitrary, and we cannot expect to have information about their first derivatives (especially if they are symbolic!) as in backpropagation, or have the time to reason about their internal structure (since we have no search).

We have been experimenting with this optimization scheme in miniature systems, as a prelude to committing to it with a full realtime multiplanner. Our first miniature merely tested the ability of the algorithm to select connections between nodes computing miscellaneous functions in order to detect symmetry in a vector of bit-values. The network was seven layers deep, and included both numeric and logical (symbolic) functions indiscriminately connected as start. The learning curve is shown in figure 2.

We are currently working on a miniature which, given the operators toggle-bit and swap-bits, will learn how to make the vector symmetrical with a minimal number of operators.

7 Determining the Content of the Nodes

AI systems that move away from "universal" methods such as search depend heavily on well-chosen and well-organized knowledge to compensate for their

limited power. Especially crucial in our variety of planner are the repertoire of plans, how they are broken up into planning fragments for storage in action nodes, and features which serve these nodes. The planner designer must steer a course between plan fragments that are too general to be served by any reasonable set of features, and plan fragments that are so specific that each one requires its own special set of features. Ideally, we would like to have moderately general plan fragments that can be cued by relatively low-level features.

How to organize and mediate between behaviors is an open question in agent-based architectures. Architectures such as the Society of Mind [Minsky 86] and Subsumption Architectures [Brooks, Connell, & Peter Ning, 88] rely upon the idea of increasingly sophisticated layers of behavior, each higher layer handling circumstances that are too subtle or complicated for the layer below. We have avoided this layering approach for two reasons. First, the low-level process model of our architecture already mediates between competing behaviors. Second, layering clouds the prospects of an implicit ideal of these systems: that behaviors can be added or altered without having to modify all other behaviors already in the system.

In looking for an organizing principle for the construction of our behavior corpus, we are more concerned with the retrieval issues discussed above. An insight that we find useful in the selection of plan fragments is the following line of reasoning:

- Plan fragments can be classified by strategy, for example, *hoard*, *assume availability of item*, or *wait until item is more accessible*.
- Strategies in turn can be organized by resource types, for example, *perishable*, *permanent*, *core*, or *self-replicating*.
- Resource types often correlate with relatively low-level features, for example *is-food* with perishable resources, *is-moving* with time-specific resources, or *is-anchored* with location-specific resources.

Consequently, there is a useful link between relatively low-level features and highly abstract characterizations of plans. To exploit this in the design of our planner, we are collecting a set of low-level features which effectively organize a list of strategy-types we would expect a forest animal to use. These features are then used to group a corpus of plan fragments. Within each group we then determine which middle-level features would be necessary to distinguish between plan fragments, and which additional features are necessary to identify possible fillers for plan preconditions. Thus we are able to produce a full specification of the features an agent would need to compute, given an environment.

8 Problems and Acts of Faith

A realtime multiplanner is capable of acting in constant time because of the severe limitations we have placed on its means of computation. Many of these are inspired by the challenges of a real, volatile world. Others stem from the need to work around the lack of a "universal" computational mechanism. Although this planner may prove to be a good set of commitments in view of the environmental demands, it is not ideal. There are several useful properties of search-based projective planners which realtime multiplanning lacks.

8.1 Labelling is Hard

Naming is hard for an agent that doesn't have labels supplied to it by the environment or tutor (e.g. "the red block"). Naming not only presumes that the agent has a category for an object it encounters, but that it has some basis for distinguishing the object from other category exemplars it may encounter, even if this is the first such object it has ever seen. This means it has to know what is unique about the object with respect to other possible exemplars of the category. Without this ability to discriminate within categories, the agent is incapable of powers such as object permanence. The agent may often begin a task with one object, be interrupted, and continue the task with another object in the same role, oblivious to the change. This is disastrous if, for example, the agent is building its shelter, stops, and resumes in another place, completing the shelter of a rival.

One possible solution lies in more detailed descriptions of contexts in a memory pools. Presumably there are some features which are not important to an object's role in the plan, but distinguish it both from other objects in the same category and from the larger scene in which it was noticed. A fairly reliable example of this kind of distinguishing feature is location.

8.2 Planning without Protections

Most conspicuous of our planner's shortcomings is a lack of protections. The extreme parsimony in use of variables, the difficulty of labelling, and the sheer number of plans that may be simultaneously active all make checking for protection violations prohibitive. Every time the planner selects a plan it is making resource commitments: effectors in the short term and objects in the environment in the long term. When the planner suspends a plan it does not forfeit those commitments. But it *doesn't* check to see if any of those commitments are violated by subsequent plan inceptions. This would be too expensive, checking every resource against every plan. It is easy to imagine an agent picking up a tool and then promptly putting it down because it saw another, and then picking up the first again.

There are two reasons why a lack of protections may not be a stumbling block. The first is a variation on the Friendly World Assumption: in most environments the range of useful plans makes little use of protections, and where protection violations lead to trouble, the environment will soon change so that only one of the plans competing for the resource is still applicable.

The second we may call the Underspecification Assumption: many goals which seem to require protections need them only because they contain too little information in their specification. The classic example, stack block A on block B and block B on block C, would be beyond our agent. The stack two blocks plan fragment would put A back on B every time the agent figured to clear B to put it on C . However, a plausible real-world alternative, stack-the-blocks-in-size-order, is easily handled by repeated executions of the put-the-biggest-free-block-on-the-pile plan fragment.

9 Acknowledgements

This work was supported in part by the Defense Advanced Research Projects Agency and monitored by the Air Force Office for Scientific Research under contract number F49620-88-C-0058, and in part by a National Science Foundation Graduate Fellowship. The Institute for the Learning Sciences was established in 1989 with the support of Andersen Consulting, part of The Arthur Andersen Worldwide Organization.

References

[Agre 88] P. Agre. *The Dynamic Structure of Everyday Life*. PhD Thesis, MIT Department of Electrical Engineering and Computer Science, 1988. Available as Report AI-TR 1085.

[Agre & Chapman 87] Philip E. Agre and David Chapman. Pengi: An Implementation of a Theory of Activity. In *Proceedings of AAAI-87*, 1987.

[Birnbaum 86] L. Birnbaum *Integrated Processing in Planning and Understanding*. PhD Thesis, Yale University Computer Science Department, 1986. Available as Report RR-489.

[Brooks, Connell, & Peter Ning, 88] Rodney Brooks, Jonathon Connell, and Peter Ning, Herbert. A second generation mobile robot, AI Memo 1016, MIT Artificial Intelligence Laboratory, 1988.

[Firby 89] R. James Firby. *Adaptive Execution in Compplex Dynamic Worlds*. PhD Thesis, Yale University Computer Science Department, 1989. Available as Report RR-672.

[Green & Swets 66] David M. Green and John A. Swets. *Signal detection theory and psychophysics*. New York: J. Wiley & Sons. 1966.

[Kirkpatrick, Gelatt & Vecchi 83] Scott Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi. Optimization by simulated annealing. *Science* 220:671-680. 1983.

[Maes 89] Pattie Maes. The Dynamics of Action Selection. In *Proceedings, IJCAI-89*. 1989.

[Minsky 86] Marvin Minsky. *The Society of Mind*, New York: Simon and Schuster, 1986.

[Tanner, Swets, & Green 56] W. P. Tanner, Jr., J. A. Swets, and D. M. Green. Some general properties of the hearing mechanism. University of Michigan: Electronic Defense Group, 1956. Technical Report No. 30.

Recognizing Novel Uses for Familiar Plans

Beth Adelson[1]
Dept. of Computer Science
Tufts University
Medford, MA 02155
adelson@tufts.cs.edu

Abstract

Analogical design and invention is a central task in human cognition. Often during the process the designer/inventor gets stuck; backs off from the problem; and only later, after having put the problem aside, discovers that some familiar plan can be used in a novel way to solve the problem. We describe a system which uses a causal *case memory* to check the side effects, preconditions, etc. of incoming events in order to model this phenomenon.

The method used makes this work relevant to case-based reasoning as well as design. It also forms a companion issue to execution-time planning.

1 Motivation

Analogical design/invention is a task which arises continually. Sometimes the task occurs in a form so simple as to pass almost unnoticed, as when picnickers find they have forgottten their knives. Sometimes the task requires intense and extended effort, as when computer engineers find they must develop a new architecture. Difficulty notwithstanding, the scenario often runs as follows: the designer finds that the tack he has been taking is not producing the desired solution; finding he is stuck, he puts the problem aside; he then recognizes in some later situation, that a familiar device, instrument or plan can be used in a novel way to solve the problem.

Problem Statement:
This scenario of the design process implies that cognitively based design systems need a mechanism that allows the recognition of situations that can contribute to the attainment of goals which previously could not be fulfilled.

The problem we are addressing is related to the problem of *execution-time planning*, in which a planning system needs to be able to recognize conditions that unblock previously blocked goals even though the system cannot predict when those conditions will occur (Hammond, 1989; Alterman, 1988; Firby, 1987; Georgeff & Lansky, 1987; Simmons & Davis, 1987). Hammond describes such a system in his work on *opportunistic memory* (1989): His system

[1]This work was funded by Carnegie-Mellon's NSF funded EDRC and by a grant from NSF's Engineering Directorate.

contains a detailed taxonomy of the conditions under which goals which were blocked for various reasons might later be fulfilled. As a result, under Hammond's model when a goal is blocked it is suspended and memory is tagged for the conditions which might allow its satisfaction. When these conditions are later encountered, becuase of the tags placed in memory, they are recognized as unblocking conditions and the goal is reactivated[2] and then pursued.

Design systems face a somewhat different problem. They need to recognize conditions that will contribute to goal satisfaction even though they can predict neither when the conditions will occur, nor what the conditions will be. In this paper we describe the mechanism we have developed for addressing this recognition problem. The mechanism is one that considers the implications (side effects, etc.) of well-known plans in order to discover their relevance to blocked goals. The effect of the mechanism is that the system is able to recognize novel uses for familiar plans.

2 The Need for Recognizing Plans During Design

2.1 Previous work: Debugging analogically acquired plans

We are developing our recognition mechanism in the context of our work on analogical learning and analogical design. We began by concentrating on analogical debugging. This work then led us to our current concern with recognition.

Our initial system (Adelson, 1989a&b) was developed to acquire plans for programming operations such as **pop**, **push** and **sort**. The plans took the form of executable models. This allowed the plans to be used in either of two ways: They could be run for purposes of debugging; or they could drive a module that generated both code and box-and-arrow drawings (Adelson, 1989a&b; Burstein & Adelson, 1987; Burstein and Adelson, in press).

As described on page 3, when the plans were being used for debugging they were passed to a mechanism that, via simulation, identified functionally analogous plan entities in the more and less familiar base and target domains. This entailed addressing: 1. The role of causal reasoning; and 2. The use of target domain knowledge in analogical learning.

When the plans were being used for debugging they were passed to a mechanism that, via simulation, identified entities in the target domain that were functionally analogous to entities in the original base domain plan[3]. This entailed addressing: 1. The role of causal reasoning; and 2. The use of target domain knowledge in analogical learning.

In addition to producing programming plans, the system has modeled Edison's description of modifying the phonograph to produce the kinetoscope and Morse's account of developing

[2]Hammond's model has the advantage of allowing goals to be quiescent during their suspension. Additionally, because of its detail, Hammond's indexing scheme results in both a high rate for hits, and a low rate for false alarms.

[3]The base and target domains are the more and less familiar domains. The reason for identifying functionally analogous entities is that one purpose of analogical debugging is to find target domain entities that should replace base domain entities in order to produce a model appropriate to the target domain.

telegraphic relay stations (Brian, 1926; Okagaki & Koslowski, 1987)[4].

Example: Morse's telegraphic relay station

The debugging behavior we have previously focussed on and the recognition behavior we turn to here are illustrated by the following example (Okagaki & Koslowski, 1987):

> Initially, in trying to transmit telegraphic signals across significant distances Morse tried the strategy of building successively stronger generators. He found however, that the signals still degraded with distance. The solution to the problem came to him in the following way. While riding on a train, he happened to look out of the window and notice a Pony Express depot, at which horses were being fed and watered. Morse realized that the *relay station* strategy constituted a solution to the telegraph problem as well.

There are two interesting pieces of reasoning in this example, recognizing that the Pony Express offers a useful analogical model; and then modifying the Pony Express model to fit the telegraphy domain. First we describe the modification performed by our system's debugger (Adelson, 1989a & in press) and then we go on to describe our recognition mechanism.

The debugger

In modifying the Pony Express model to fit the telegraphy domain, the system determines, via simulation (Schank & Riesbeck, 1989; Falkenhainer, Forbus & Gentner, 1988; Hammond, 1989a; Simmons & Davis, 1987), the causal effects produced by the use of horses traveling between relay stations (the information in the message can successfully be sent a long distance as a result of the medium of conveyance being repeatedly refreshed at each leg of the journey). In an attempt to construct a mechanism that will reproduce this effect in the target domain, the system again uses simulation to identify already existing appropriate pieces of mechanism in the target domain; ones that could be used in a larger schema for repeatedly refreshing and sending information short distances. This results in the system producing a model in which the existing generators are arranged in series at appropriate intervals.

3 Recognizing the analogy

As mentioned above, in order to produce the model of generators arranged in series, Morse must first recognize that the relay station strategy employed by his competitors can serve as a model for the telegraph problem. We propose a mechanism that accounts for this class of recognition phenomenon, in which it is noticed that a familiar case in memory; a plan, operation or device, can serve a purpose that is radically different from those which it has previously served. In essence, the mechanism is one in which the changing environment is examined for opportunities which allow the fulfillment of blocked goals (Birnbaum & Collins, 1984; Birnbaum & Selfridge, 1981). In this scheme, a blocked goal is placed on a list with other goals that have also been suspended because they were blocked. The system then sets about to parse any incoming plan in order to determine if the plan can unblock it[5].

[4]Although these may not be completely accurate accounts of particular discoveries, they persist because they are accurate accounts of our *experience* of the discovery process.

[5]Along with others any others already on this list.

At a very general level, under this model Morse realizes that the relay station strategy will meet his need because as he backs off from his plan of building stronger generators he starts looking at the features of other plans and assessing their utility in unblocking his goal. That is, when he encounters the Pony Express station he notices that it has the effect of sending messages long distances and so recognizes it as a plan that meets his goal.

The next section provides a detailed description of how this phenomenon is captured in our system. We begin with an overview of the system's function.

4 The Recognition Mechanism

4.1 System Function

Our system is designed to monitor the environment for plans which will be useful in satisfying blocked goals. In doing so the system takes the following steps:

1. Detecting and saving blocked goals:
   When the system is unsuccessful in meeting a goal, the goal is placed on a BLOCKED GOAL LIST.

2. Monitoring the environment for opportunities:
   The system then begins to evaluate events in the environment[6] in terms of their relevance to the blocked goal. In order to do so the system uses an EVENT LEXICON which can be thought of as a memory for causal relations. The EVENT LEXICON stores features of previously encountered events such as intended effects, side effects, etc.

   Because the system is looking at these effects to see if they fulfill a blocked goal, and therefore should be reproduced, each effect has a pointer to a plan which if run would give rise to the effect. That is, an event may produce a side effect which matches the desired effect of the blocked goal and if it does the system can reproduce the side effect in order to meet the blocked goal.

   The monitoriing process has two parts:

   (a) Parsing incoming events:
   When an event occurs, the system retreives the representation of that event from the EVENT LEXICON, instantiates it and places the instantiated representation in the system's short term memory.

   (b) Testing potential opportunities:
   The system now matches the features of the retrieved event representation against the blocked goal in order to see if some feature of the event will help in unblocking the goal (Hammond, 1989). If a match is found, the plan indexed under the feature is retreived.

[6] Events are provided as input to the system.

3. Seizing the opportunities:
At this point the goal is removed from the BLOCKED GOAL LIST. The plan satisfying the goal can now be instantiated and run. If the plan, when run, meets the goal it will be indexed under the goal in the system's PLAN MEMORY.

4.2 The Morse Example

In the Morse example this translates into the following steps:

1. Detecting and saving blocked goals:
The system is given the goal shown just below, of sending a telegraph message over a long distance:

```
(GOAL: send
       (OBJECT:  message
(TYPE: telegraph signal))
       (DISTANCE: >1,000 mi.))
```

The system finds that the goal cannot be satisfied by any of the existing plans in the system's PLAN MEMORY. The goal is therefore placed on the BLOCKED GOAL LIST.

2. Monitoring the environment for opportunities:
The system is then given a set of inputs representing the objects and events Morse encounters during the train ride: The indifferent dinner in the dining car, the elaborately feathered bonnet of his compartment mate, and the Pony Express relay station which he sees as he looks out the window[7].

   (a) Parsing incoming events:
   The system retreives its representation of the events from its EVENT LEXICON. The representations are instantiated and placed in the system's short term memory.

   (b) Testing potential opportunities:
   The system then finds that SIDE EFFECT1 in the representation for the Pony Express matches the blocked goal.

   Representation for Pony Express:

```
(EVENT: Pony Express
(GOAL: deliver mail)
(PRECONDITIONS: (...)...(...))
(SIDE EFFECTS:
       (SIDE EFFECT1: send
         (OBJECT:  message
```

[7]Not all inputs are given equal attention. We discuss the differences in attention given to different inputs in the context of the system's attention mechanism in Section 4.3 below.

```
(TYPE: mail))
                    (DISTANCE: >1,000 mi.)
                    (CAUSED-BY-PLAN: relay stations)))

(INTENDED EFFECTS: (...)...(...)))
```

That is, looking at the sbove EVENT LEXICON representation of the Pony Express, the system determines that through the use of relay stations, the Pony Express has the effect of sending messages across long distances.

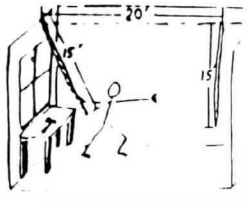
3. Seizing the opportunities:
   The system retreives the RELAY STATION plan indexed under SIDE EFFECT1 and passes it to the system to be run[8]. The plan is found to satisfy the goal and so it is indexed under the goal in the system's PLAN MEMORY.

4.2.1 Range of Examples: Einstellung and Archimedes

Our system has also been used to model a result representative of the results obtained in the classic gestalt *einstellung* experiments.

The example is as follows (Meyer, 1977):

Experimental Situation: A subject is shown into an otherwise bare laboratory, in which two cords are suspended from a ceiling and a hammer has been left on the windowsill. The subject is asked to tie the ends of the two cords together, however the distance between the cords is great enough that the ends of the cords cannot be brought together simply by grasping the first one and walking towards the second.



Result: The problem can be solved by tying the hammer to the end of the first string and then using the hammer as a pendulum to swing the first string towards the second.

Often subjects do not hit upon the solution themselves, In these instances the experimenter walks by one of the strings and 'accidentally' sets it in motion. This hint invariably allows the subject to solve the problem.

Our noticing mechanism, in modelling the above result, offers an account of the processes behind the insights which occur in these experiments. Under our model, at the outset of the experiment no plan for tying ropes which are more than an arm's length apart resides in memory. The subject gets stuck and the goal is placed on the BLOCKED GOAL LIST. When the rope swing occurs an underlying representation of the event is retreived. This is where the insight occurs; the representation of the rope swing contains bringing the ropes together as a side effect. Because this side effect matches the desired effect the rope swing is perceived as a solution to the problem. The retrieved representation also points to the

[8]Because the retreived plan is an analog to the one actually being sought it will be modified by the system's debugger before it is run.

'pendulum plan' which can reproduce the effect. When the plan is retreived, the hammer is seen as an appropriate weight for the pendulum.

This account includes an explanation of how in these situations problem-solvers come to see the environment differently after they have backed off from the problem-solving (Gick & Holyoak, 1980, 1983).

As a third example, and one which also illustrates the issue of attention, our system models Archimedes' realization that he could calculate the volume of the king's irregularly shaped crown by submerging it in water. Initially, because the system only has plans for calculating volumes of regularly shaped objects, the system places the goal of 'calculating the volume of the irregularly shaped crown' on the BLOCKED GOAL LIST. The system is then given 'Archimedes' bath' as an input event. None of the side effects of bathing: the cooling of the water, the increase in the volume of the tub's contents, etc., match the blocked goal of 'calculating the volume of the irregularly shaped crown'. However, the side effect of the increase in the volume of the tub's contents is treated as promising because it is relevant to the blocked goal's concern with volume. The system therefore pays more attention to it in the form of further processing: The system looks in EVENT MEMORY for the effects of increasing the volume of the tub's contents. It finds that one side effect is that the volume of the object that causes the increase can be obtained by calculating the difference between the new volume and the original. This side effect matches the blocked goal and is treated as a solution to the problem.

4.3 The attention mechanism

What attentional phenomena do we want a discovery system to capture and how should we implement the mechanisms giving rise to them? We have addressed this problem to a limited extent. At minimum we would would want to explain: Why Orville Wright but not Samuel Morse would notice a feathered bonnet; and why Morse certainly and Wright only perhaps would notice the Pony Express[9]. Although it is for a different reason, we also want to explain why Archimedes would notice the implications of the rise in water level but would not bother to trace the implications of the water cooling. In summary, we want to explain the fact that things which are either psychologically charged or conceptually relevant may receive more attention.

In order to model these effects we have implemented the system so that it does more retrieval if an event looks relevant to a blocked goal in either of these ways. That is, if an event mentions a competitor or a mentor the features not only of the event, but also, if needed, of the event's effects will be retrieved from memory and examined by the system. Additionally, as illustrated in the Archimedes example, if a feature of an event is conceptually relevant to the blocked goal the features of the feature will be retrieved and examined. As a result, we have made a first pass at attentional issues.

[9]At the time the Pony Express was Morse's competition.

5 Summary and Relevance to other Research

Analogical design and invention is a central task in human cognition. In our system a causal *case memory* is used to check the side effects, preconditions, etc. of incoming events, this allows the system to put familiar plans to use in novel ways. This can account for the reasoning that often is needed in analogical design and invention.

In addition to the relevance of our work to case-based reasoning, our design task is a companion task to execution time-planning. In execution-time planning, a planning system needs to be able to recognize conditions that unblock previously blocked goals even though the system cannot predict when those conditions will occur (Hammond, 1989; Hammond, Converse, & Marks, 1988; Kolodner, Simpson, & Sycara-Cyranski, 1985). Design systems need to recognize conditions that will contribute to goal satisfaction even though they can predict neither when the conditions will occur, nor what the conditions will be. It is our hope that the mutual relevance of these tasks will motivate work on the integration of design, cased-based reasoning and planning systems.

6 References

Adelson, Beth. Cognitive modeling: Uncovering how designers design. *The Journal of Engineering Design.*. Vol 1,1. 1989.

Adelson, B. The role of model-based reasoning in analogical learning. *Proceedings of the IJCAI-89 Workshop on Model-Based Reasoning.*

Adelson, B. Characterizing the nature of analogical reasoning. In Design Theory and Methodology. M. Waldron (Ed.). Springer-Verlag. NY. In press.

Alterman, R. Adaptive planning. *Cognitive Science*, Winter, 1988.

Birnbaum, L. and Selfridge, M. In *Inside Computer Understanding.* 1981. Erlbaum, Hillsdale:NJ.

Birnbaum, L. and Collins, G. Opportunistic Planning and Freudian Slips.

Brian, George. Edison: *The man and his work.* Knopf: Garden City, NY. 1926

Burstein, M. and Adelson, B. Mapping and Integrating Partial Mental Models. *Proceedings of the Tenth Annual Meeting of the Cognitive Science Society*, 1987.

Burstein, M. and Adelson, B. Analogical Reasoning for Learning. in *Applications of Artificial Intelligence to Educational Testing.* R. Freedle (Ed.) In press. Erlbaum: Hillsdale, NJ.

Firby, R.J. An investigation into reactive planning in complex domains. In: *Proceedings of the Sixth National Conference on Artificial Intelligence*, Seattle, Washington, July 1987.

Georgeff, M.P.& Lansky, A.L. Reactive reasoning and planning. In: *Proceedings of the Sixth National Conference on Artificial Intelligence*, Seattle, Washington, July 1987.

Hammond, K. Opportunistic Memory. *Proceedings of the Machine Learning Workshop*, 1989.

Hammond, K. *Case-based planning: Viewing planning as a memory task.* Academic Press, Cambridge, Massachusetts, 1989.

Hammond, K., Converse, T., & Marks, M. Learning from opportunities: Storing and reusing execution-time optimizations. In: *Proceedings of the Seventh National Conference on Artificial Intelligence*, St. Paul, Minnesota, August 1988.

Kolodner, J. In *Proceedings of the Seventh Annual Conference of the Cognitive Science Society.* Boulder, CO: Cognitive Science Society, 1985.

Kolodner, J.L., Simpson, R.L., & Sycara-Cyranski, K. A process model of case-based reasoning in problem solving. In: *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, Los Angeles, California, August 1985.

Okagaki, L. and Koslowski, B. Another look at analogies and problem-solving. *Journal of Creative Behavior*, 1987, *21*, 1.

Schank, R. and Riesbeck, C. *Inside Computer Understanding.* Erlbaum: Hillsdale, NJ. 1981.

Simmons, R., & Davis, R. Generate, test, and debug: Combining associational rules and causal models. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, Milan, Italy, August 1987.

Planning to Learn

Lawrence Hunter
National Library of Medicine
Bldg 38A, MS-54
Bethesda, MD 20894
HUNTER@NLM.NIH.GOV

Abstract:
The thesis of this paper is that learning is planful, goal-directed activity – that acquiring knowledge is intentional action. I present evidence that learning from one's experiences requires making decisions about what is worth learning, regardless of the specific mechanisms underlying the learning or of the degree of consciousness or automaticity or level of effort of the learning. Decisions about what is worth learning are the expressions of *desires about knowledge*. I then sketch a theory of whence desires for knowledge arise, how they are represented, and how they are used. A taxonomy of learning actions is also proposed. This theory has been partially implemented in two computer models, which are briefly described.

Introduction
The central claim of this paper is that learning is planful, goal-directed activity – that acquiring knowledge is intentional action. If true, this thesis raises the relevance of both action psychology and AI planning to theories of learning. Action psychology (e.g. Tolman (1932) or Frese & Sabini (1985)) is based on the ideas that human behavior is directed towards the accomplishment of goals, that it is directed by plans, that those plans are hierarchically arranged, and that background knowledge and the environment interact in the creation and execution of plans for the guidance of action. As Frese & Sabini observe (p. xxiii) such a view is no doubt a good model of some behaviors and a poor model of others. Here I will try to demonstrate that this view plays an important role in understanding how people and machines learn from complex experiences.

Machine learning research has been dominated by the view that learning is a kind of search (see, e.g., Langley, Gennari & Iba, 1987). I believe that *planning* forms a better foundation than search for learning. In AI, planning is a set of techniques for selecting and combining actions to achieve explicit goals; see, e.g. Sacerdoti, (1971) or Charniak & McDermott (1985). The AI work originally focused on decomposing abstract goals into "primitive" actions expected to achieve the goal given specific resource limitations (such as time and energy). It has since evolved to consider issues such as uncertain environments, managing complex action over time, revising plans during their execution, taking advantage of unexpected opportunities, avoiding unexpected dangers (Birnbaum 1986), and "situating" the planner in the environment generally (Chapman 1985). The goals and actions considered in AI planning are grounded in the physical world. Domains tackled by AI planners thus far include stacking blocks, scheduling deliveries, creating recipes, suggesting battle tactics, and many others. Applying this research to the task of learning requires mapping work on planning in physical domains onto planning in mental domains. That is, a theory of planning to learn must describe the origins and nature of goals to learn, the actions that can be taken to learn, the mental and physical resources those actions contend for, the aspects of the environment that a learner must take into account when planning, and a process for selecting among and combining actions to accomplish the goals, given limited resources and the embedding of the learner in its environment. This paper sketches such a theory.

Learning Requires Decision-making
What is the evidence that learning is goal driven? There are several converging lines of argumentation. The simplest is the widespread use of goal/plan language by people discussing the acquisition of knowledge. "Inquiring minds *want to know*," "prerequisites for courses," and "research strategies" are references to goals, action preconditions and plans (respectively) for gaining knowledge. People can easily enumerate many different kinds of plans for learning, and the situations in which they are appropriate. Some general examples of common plans to learn include: ask an expert, look it up in a book, watch for it in the newspaper, try it yourself and see, wait until it happens again, or even run a scientific experiment. People are also capable of generating a variety of very specific knowledge acquisition plans for commonly occurring knowledge acquisition goal types. For example, when asked how to find Marvin Minksy's home phone number, one person suggested seven different potential plans: try Boston directory assistance, call MIT and ask in the CS department, ask some other senior AI

researcher, try the AAAI membership directory, send him email, ask the reporter who interviewed him for Science recently, or call the publisher of *Society of Mind* and ask them. He was also able to rate the likelihood of success of each of these plans, and could suggest general principles about how to get similar information in various circumstances. The widespread use of goal/plan language in describing learning, and the ability of people to easily generate plans when given statements of desired knowledge is, for now, merely anecdotal. A stronger line of argument comes from an analysis of the combinatorics of inductive learning from complex experience.

Inductive inference is limited in at least two ways. First, a recent proof by Deitterich (1989) demonstrated that learning algorithms, very broadly defined, can evaluate only a small proportion of the hypotheses compatible with the experiences they have. That is, there are far more hypotheses consistent with experience than can be distinguished among using that experience. A brief sketch of the proof is as follows: Consider inferential effort required to learn from experience, even in the drastically simplified situation assumed in the PAC-learnablity literature (Valient 1984). This simple task used for computational complexity analyses, involves a learner attempting to induce a subset h of the set of all Boolean n-tuples by processing m distinct examples, where an example is an n-tuple labelled as to its membership in h. There are $2^{(2^n-m)}$ hypotheses that are consistent with the m examples seen so far. That is, each of the 2^n-m possible n-tuples that are not examples could be in h or not. The size of that hypothesis space grows *double exponentially* in the complexity of the experiences. Information in the experiences grows only exponentially in their complexity. Learning by searching a space whose size is proportional to the space of hypotheses consistent with experience is therefore intractable.

The second problem with inductive algorithms stems from the fact that all known machine learning algorithms require time proportional to the number of features that can appear in their inputs, that is, the number of features they can "perceive." Learning systems that take time dependent on the number of perceptable features in the universe will be unable to account for human behavior, and are unlikely to be adequate for applying machine learning to desirable technological tasks (e.g. analyzing data from the human genome project). This is not to say that these algorithms may not prove to explain a *portion* of human learning, but alone they cannot form a sufficient theory.

To see how daunting the problem of learning algorithms that take time proportional to the complexity of their experiences is, consider the complexity of human experience. An order of magnitude estimate of the amount of afferent (incoming) information can be sketched easily: approximately 10^{10}–10^{12} nerve cells in the body, conservatively 1% to 10% of them sensory nerves, each capable of carrying between 100 and 1000 bits per second. Combining to find a very conservative lower bound, humans routinely handle at least $10^{10} \times 10^{-2} \times 10^{2} = 10^{10}$ bits per second at their sensory surfaces over their entire (waking) lifetimes, and quite possibly as much as 10^{13} or more bits per second. There is simply more information available in human-like experience than inductive algorithms can handle.

Machine learning theories of induction are not generally proposed as cognitive models of human learning generally (although they are sometimes advanced as models of category formation or categorical perception), and this is one of the reasons. The computational complexity of searching the hypothesis space generated by experiences anywhere near as complex as human experience would take computational power far exceeding even generous estimates of the computational power of the human brain. And since people use extensive background knowledge in learning, the complexity of the search space of is further increased by the interaction between experiences and all of memory. Schank, Collins and Hunter (1986) argued that inductive category formation approaches fails on other pragmatic grounds as well.

It is important to note here that parallelism does not provide a simple solution to these problems. The number of hypotheses consistent with a set of experiences grows *double exponentially* in the complexity of the experiences. Straightforward parallelization of the search through this space would require a number of processors exponential in the complexity of inputs to make this even an exponential time task. For comparison, current artificial neural net technology (e.g. Rummelhart, McClelland & the PDP group, 1986), typically uses a number of processors proportional to the complexity of the inputs; there are typically far fewer hidden nodes than input nodes in these networks, since too many hidden nodes reduces the usefulness of the learning these networks do.

Since the hypothesis space generated by human-like experience is far too large to be searched, even sublinearly, and all induction algorithms take time dependent on the complexity of their inputs, what

can people be doing when they learn from their experiences? The inescapable conclusion is that they must somehow drastically restrict the space of hypotheses that they consider during learning. How?

A key first step is the transformation of inputs to more compact representations of experience, capturing the "important" aspects and reducing the amount of "irrelevant" information. This process is hardwired perceptual processing and is likely to be automatic and fast (Fodor, 1985). However, even the transformed representational space will be quite large in systems capable of human-like behavior. It is simply the case that people are sensitive to a very large number of potentially relevant stimuli, and that this large number of "features" is overwhelming to known learning algorithms. So what can be done to restrict the size of *this* space to manageable proportions?

Existing machine learning methods have restricted the size of this space by applying *inductive biases*, e.g. Utgoff (1986), or by *a priori* limitations on the structure of the hypothesis space, through, for example, the use of decision trees or neural networks. These approaches can be considered *syntactic*, in that they constrain the *form* of the hypotheses considered, rather than their content.

I propose that the method of restricting potentially learnable hypotheses for both people and effective machine learning systems should be *content*-based. Explicit characterizations of *desirable knowledge* provide a principled method for restricting the realm of experience and background knowledge considered in learning, and thereby the size of the hypothesis space that must be considered. Having goals specifying what (kind of) knowledge is desirable provides a significant advantage for systems trying to learn from very complex experience.

Why does having explicit knowledge acquisition goals provide an advantage? The idea is to exert the broadest effective top-down constraint on the space of possible concepts to learn. Bidirectional inference, i.e. the ability to use top-down constraints (in this case, goals) as well as bottom-up information (here, processed perceptual data), is the most effective known technique for reducing the size of a space that has to be searched to find desired concepts (Birnbaum 1986).

This claim is consistent with a large body of psychological research on goal direction in selection of focus of attention, particularly from social psychology. Zukier's (1986) review concludes: "Experimental studies have clearly demonstrated that a person will structure and process information quite differently, depending on the future use he or she intends to make of it. Information integration clearly is preceded by future-oriented decision-making processes, which guide data selection and the choice of an appropriate strategy or mode from among the several that are available," (p. 495).

Hoffman, et al (1981) demonstrate that different goal orientations (e.g. "form an impression of a person in the following story" or "remember as much as you can from the following story.") may influence not only to the use of different representations, but also the selection and use of different kinds of information processing. Although the goal orientations tested in that work are quite abstract, they significantly constrain the space of hypotheses consistent with the experimental materials. Srull & Wyer's (1986) results, although divergent in important respects from those of Hoffman, et al, also provide evidence that different goal orientations have a strong effect on learning. These results bear an interesting relationship to the one of the implications that Deitterich (1989) draws from his proof about machine learning algorithms (p. 128):

> [D]ifferent classes of learning problems may call for different learning algorithms. An important problem for future research is to attempt to identify relationships between types of learning problems... and types of hypothesis spaces....

That is, the combinatorics of learning require the selection of learning methods that are appropriate to particular kinds of problems, and goal orientation clearly effects the results of learning. This convergence of evidence from both psychological studies and from computational complexity analysis in machine learning suggests a hypothesis about the control of learning: *Goals about what would be desirable to learn are central to making required decisions about what and how to learn.*

Related Previous Research

Other cognitive theories have also included reference to desires for knowledge, although there are significant differences between those prior theories and the current claims. For example, consider the D-KNOW (delta-knowledge) class of goals, which are part of the conceptual dependency representation proposed by Schank & Abelson (1977). They are goals to "change knowledge state," i.e.

to learn something. Examples of D-KNOW goals were to find out the location of food (in order to go to it and then eat it) or to find out the price of an item (in order to buy it). The generation of D-KNOW goals was always tied very specifically to a physical supergoal (e.g. satisfy hunger), and were not mentioned in the author's later theories of learning (e.g. Schank, 1982). Other theories, particularly from the animal learning psychology literature, have proposed general motivations to learn: a "will to perceive" (Thorpe), a "motivation for learning" (Thacker), and a "search by an information hungry organism" (Pribram – all reported in Livesey, 1986, p. 20-21), but these theorists did not propose any specific desires, just diffuse drives. Social psychologists have used various "goal orientations" as explanatory phenomena in theories of attention, recall and judgement, which are close in spirit to the goals to learn proposed here. However, social psychologist's goal orientations are generally specified at a very abstract level (e.g. "Form an impression," or "make predictions"), and as Zukier's (1986) review notes, "In general, however, little systematic research is available on goal orientation in inference, and no comprehensive taxonomies of 'middle-level' or concrete goals have emerged from these studies."

Also related to the current claims is the work of Horvitz, et al, (1989). They present a calculus for deciding when to do more inference (versus when to act) in medical decision making. Although based on highly idealized functions for estimating the expected value of additional inference (in their model, inference includes data gathering), it provides an attempt to model content-based decisions about *when* it is worthwhile to acquire knowledge. Although their model does not specify *what* is worth learning, it may be useful in deciding whether it is worth learning at all, potentially reducing the size of the potential hypothesis space to zero. Minton (1988) also proposes a model of judging whether it is worth learning, although his model involves computing the effect of learning on future performance *after the new concept is formed*, and is hence not useful for constraining the hypothesis space.

Both failure-driven (e.g. Schank 1982) and success-driven (e.g. DeJong & Mooney, 1986) computer models of learning posit very direct connections between experiences and (implicit) desires to learn. In these systems,. the learning always takes place at the time of the failure (or success), and anything can be learned at that time is learned. A system that plans to learn may generate learning goals as a result of a success or failure, and may (or may not) be able to achieve those goals at that time. The role of failure (or success) in planning to learn systems is to identify knowledge that is worth pursuing, not (necessarily) to signal the time when knowledge can be acquired; they are failure (or success) *motivated*, not failure (or success) *driven*.

In the remainder of this paper, I sketch a theory of the origins and uses of explicit goals about what to learn. Some aspects of this theory of knowledge acquisition goals and knowledge acquisition planning are presented in greater detail in Hunter (1989, 1990a and 1990b).

Learning Goals

In order to make learning computationally feasible, learners must have goals specifying what they wish to learn, which are used to constrain the space of possibly inducible concepts. How are these goals represented? Where do they come from? How do they influence the learning process?

Desires about knowledge can be represented in at least two distinct ways. The first representational format is based on a description of the function that the desired knowledge will fulfill. These are generally stated as "desires to know how," such as the desire to know how to distinguish between mushrooms and toadstools, or how to recognize a potential good deal in the real estate ads. The other representational format is a description of the relationship of the desired knowledge to a set of existing knowledge; for example, the desire to know the capitols of all 50 states, or the names of your boss's children.

The relation-to-other-knowledge representation of learning goals is similar to Lehnert's (1978) work on representation of questions in natural language understanding. In her computer model, questions were represented by the same knowledge structures that held memories, but with some of the unfilled slots in those structures identified as the subject of a question. It is also possible to use her representational strategy for the internal representation of knowledge acquisition goals: goals can be represented as pointers to certain unfilled slots in memory structures. Ram (1989) presented a theory where relation-to-other-knowledge representations of questions were used to drive natural language understanding. Many of Ram's results apply to the design of knowledge acquisition planners generally.

Where do unfilled slots in memory structures come from? In general terms, they come from the incomplete instantiation of knowledge schemas. In order to generate relation-to-other-knowledge goal representations, a learner must have some knowledge of the structure of its knowledge. Consider a simple example: in order to represent the goal to find the capitols of all 50 states, a learner must know that states have capitols. That knowledge implies that the representation for each state will have a "Capitol" slot, and (presumably) the values for some of those slots are unfilled. Those unfilled slots can be the subject of a desire for knowledge. That is, the relation-to-other-knowledge representation of a goal to learn is the result of the application of some knowledge about the structure of knowledge to form a characterization of a gap, which is a representation of desired knowledge. Ram (1989) presents a much more detailed mechanism for generating these kinds of goals during the process of understanding natural language.

The other form of knowledge goal representation, based on the function of the desired knowledge, arise from inferences about knowledge useful for particular tasks. The knowledge underlying these inferences provide mappings from desired performance to desired knowledge. The resulting representations specify the processes in which the knowledge will be used, and the role that it will play in those processes. Another simple example: in order to do diagnosis, one must know (a) the kinds of things that can effect the behavior of a system and (b) methods for distinguishing among alternative potential causes of the to-be diagnosed behavior. When the need to diagnose, say, computer disk-drive failures arises, that high level knowledge can be used to generate goals to find out about the ways disk drives can fail and how to distinguish among them. The general knowledge must identify where in the diagnostic process the desired information will be used and for what, so that when it is found the information can be stored in the appropriate place for later use. See Hunter (1989) for detailed examples of the generation and representation of this kind of knowledge goal in a diagnosis domain.

Planning to Learn

The generation and representation of goals to learn is only the beginning of the learning process. The theoretical justification for generating them depends on their effectiveness at constraining combinatorics of learning from complex experience. I indicated briefly that learning should involve bidirectional inference: top-down, from learning goals *and* bottom-up, from experiential data. How can this be accomplished?

The idea is to use AI planning techniques for making decisions about which learning actions should be taken in what order to achieve the knowledge goals of an actor situated in the world. Generally speaking, these decisions are based on knowledge about available resources, knowledge about actions and knowledge about the current state of the world (including the actor's current knowledge state). Planning to learn is much like other kinds of planning, so here I will just try to describe the kinds of knowledge about resources, actions and states of the world that are necessary for planning about learning, rather than describe the process itself. The source of following characterizations are the computer models IVY and INVESTIGATOR. IVY was primarily an exploration in deciding what was worth learning, and INVESTIGATOR focuses on how to learn given a set of learning goals. They are described in detail in Hunter (1989) and (1990a), respectively.

Learning Actions

The actions that people take to acquire knowledge span a tremendous range, from looking up an answer in a reference book to designing and running scientific experiments. In order for a planner to select actions appropriate to goals, the actions must be annotated with the resources that they require, preconditions to executing the actions and expected outcomes of the actions (and perhaps information about possible alternative outcomes and relative probabilities of the alternatives). Here we will consider some of those actions and their representations.

In a system capable of taking a large number of possible actions, hierarchies of action classes can improve the combinatorics of the planning process. Classes of knowledge acquisition actions are, in effect, hypotheses about the component cognitive processes involved in learning. IVY and INVESTIGATOR, two computer models of planning to learn, use very different actions and learn from very different sorts of data, but their actions can nvertheless be grouped into four clearly defined classes:

• *Finding examples of specified phenomena.* This class of actions maps abstract characterizations (phenomena) to specific instances (examples). In IVY and INVESTIGATOR, these actions fall into two

subclasses: explicit data gathering and perceptual processing. INVESTIGATOR maps characterizations to instances by doing various kinds of database lookups. IVY works "perceptually," checking for inputs that match a desired characterization while doing its main task of diagnosis. Both subclasses require as preconditions representations of the desired phenomena that can be used to acquire or recognize examples. In addition, the data gathering actions require access to the sources of data. Each particular action further specifies the general preconditions; e.g. to look up bibliographic records from Medline^{tm}, INVESTIGATOR must form a query in the Elhill retrieval language and be able to open a network connection to the Medline^{tm} server. The resources consumed by this class of actions (see below for a discussion of learning resources) are the time it takes to find the desired example, and the memory required to store the found examples. The expected time to find examples perceptually can be large (i.e. you do not know when you will find what you are looking for). The expected amount of memory required for some database searches can also be large.

• *Grouping examples.* The actions in this class create collections of related examples. Subclasses of these actions include finding similar examples (using various metrics), clustering examples into equivalence classes, and building hierarchical clusters. The precondition to this class of actions is a collection of examples. For example, given a genetic sequence (say, retrieved from a database) INVESTIGATOR can use sequence matching algorithms to find other genetic sequences it knows about that are similar to it. Very few resources are required for this action. INVESTIGATOR can also use Cheeseman's (1989) Bayesian classification program Autoclass II to divide a collection of objects into clusters. That action requires significant amounts of time and CPU cycles. Some other grouping actions (e.g. hierarchical clustering) also require an applicable distance metric as a precondition.

• *Generating Abstract Characterizations of Groups.* This diverse class of learning actions includes many of the techniques traditionally associated with machine learning: concept formation, statistical analyses of collections of examples, and forming explanations of phenomena. This class of actions maps from a collection of examples and a collection of existing abstractions to a new abstract characterization of the collection of examples. INVESTIGATOR's abstraction actions so far include an inductive category formation algorithm (which generates conjunctive definitions from groups of positive and negative examples) and ANOVA algorithms for determining the statistical features of collections of examples. These actions do not use existing characterizations: they map directly from a set of examples to an abstract characterization. Any learning method that uses domain knowledge uses both examples and existing abstractions (the domain knowledge) to form new abstractions (e.g. explanations of the examples). Although the actions in this class vary a great deal, their preconditions and expected results are similar enough so that it is possible to formulate useful planning knowledge that refers to this general class.

• *Mapping Abstract Characterizations from One Group to Another.* This class of learning actions transfers characterizations from one group to another. INVESTIGATOR currently has only one action in this class: a marker passing method for mapping a distinction in one hierarchy into another. The preconditions are two hierarchies, a distinction in one, and a mapping between the leaves of the hierarchies. This action was used to map a distinction in a taxonomy hierarchy (grouping organisms into classes) into a protein family hierarchy. The individual proteins were labelled with the organism that they came from, i.e. there was a map from the leaves of the protein hierarchy to leaves of the taxonomy hierarchy. Executing the action found protein families that were associated with specific taxa. Although not implemented in either program, this class also contains all of the learning actions involving analogy, as well as methods for mapping knowledge across dissimilar groups of examples (e.g. intersection search).

Individual learning actions are rarely able to satisfy knowledge acquisition goals; they must be assembled into sequences of actions – into plans to learn. In INVESTIGATOR, the generation of learning plans is done by top-down subgoal decomposition. Decomposition rules embody knowledge about what the various knowledge acquisition actions and classes of actions are good for. Knowledge acquisition goals are transformed into subgoals, and the subgoals are further decomposed until the process bottoms out in specific knowledge acquisition actions whose total resource consumption does not exceed preset limits, and all of whose preconditions can be satisfied. IVY did not do its own subgoal decomposition, but used programmer assembled stereotypical plans. However, INVESTIGATOR is strictly top-down, and cannot currently take advantage of unexpected opportunities, whereas IVY was able to select among potential knowledge acquisition plans based on opportunities detected during routine performance.

Work is currently underway to make INVESTIGATOR's planning more sensitive to its situation, creating a mechanism for exploring data and partial results in a more bottom-up, opportunistic fashion.

Learning Resources

With unlimited resources, planning is trivial. Unfortunately, there are always limits. Physical planners have to manage resources like energy, money and time. Planning to learn is similarly constrained, although the resources are different. In particular, learners have limitations on the amount of memory they have and on the amount of time they can spend on inference. Programs like INVESTIGATOR may also have limits on the amount of network traffic they generate. Traditional physical planning resources may also come into play, e.g. database access may cost money. Planners may have strict limits on resource consumption, or may merely try to avoid waste. INVESTIGATOR has estimates of the resources each of its actions will consume, and selects among alternative plans for accomplishing a goal by minimizing the resources consumed. It can also reject plans that exceed preset limits, e.g. would take thousands of CPU hours or gigabytes of storage.

For INVESTIGATOR, memory and CPU cycles are the constraining resources. Some of its knowledge acquisition actions are directly annotated with a formula for estimating the resources consumed. The resources consumed by others can be inferred from generalizations associated the class of which the action is a member. For example, grouping examples is assumed to take time and memory proportional to the number of examples. The Autoclass II grouping method overrides those defaults, specifying that it takes a large amount of time initially, plus time proportional to the number of examples times times the complexity of the examples times the number of expected classes). Because INVESTIGATOR tries to conserve resource consumption, Autoclass is not used unless the other grouping methods fail or unless its particular kind of output is a prerequisite for some other action.

The question of managing resources in learning raises the issue of learning over time. Existing machine learning research has focused on learning from a particular dataset. Conversely, human-like learning occurs over an entire lifetime. Learners need to decide not only whether and what to learn, but when to learn. IVY is able to keep "questions in the back of its mind," in the form of pending learning plans, which are executed as opportunities arise. A more sophisticated planner might manage a complex and interacting set of learning goals, making decisions about when to pursue a particular goal, based on its relationship to the program's other learning and performance goals and on on the current state of the world.

Conclusion: Decision-making in Learning

The space of possible lessons from experience is so large that it is combinatorically implausible to learn them all. A learner situated in a complex world must therefore make decisions about what is worth learning. The results of these decisions are explicit (although not necessarily conscious) goals about the knowledge a learner desires. Learning is not a passive process: learners act in order to learn. Their goals can be used to direct the selection of the actions taken.

Planning is decision-making based on expectations about the outcomes of actions. Effective learning decisions require knowledge about the kinds of actions that can be taken to acquire and transform knowledge, and the resources that those actions consume. Knowledge about learning actions used in planning includes information about the prerequisites for taking an action and about its expected results. Algorithms modeled on AI planners in physical domains can be used to select courses of action that can be expected to yield the desired results under resource constraints. Unlike physical planning domains, the limiting resources in learning are often inferential effort (CPU cycles for computer systems) and memory capacities.

The evidence for viewing learning as a planning process comes from both combinatoric arguments and empirical results in action psychology. This view raises a variety of issues not traditionally dealt with in the machine learning or cognitive psychology literature: How do learners come to have specific desires about knowledge? What kinds of desires to people have about knowledge? For example, can they fear specific kinds of knowledge? How do large numbers of goals to acquire knowledge interact? Can they interfere with each other the way physical goals can? How are they prioritized? The planning process raises questions of its own: How can learners recognize unexpected opportunities to learn? What are the actions that people take to learn? How are those actions organized and selected among? What do people know about the learning actions themselves, and can new actions be learned? Answers to these questions await future research.

References

Birnbaum, L. (1986). Integrated Processing in Planning and Understanding. PhD. thesis, Yale University, New Haven, CT. (Technical Report YALEU/CSD/RR#489)

Chapman, D. (1985). Planning for Conjunctive Goals. (Technical Report 802). Boston, MA: MIT AI Laboratory.

Charniak, E., & McDermott, D. (1985). Introduction to Artificial Intelligence. Reading, MA: Addision-Wesley.

Cheeseman, P., Kelly, J., Self, M., Stutz, J., Taylor, W. & Freeman, D. (1988). AutoClass: A Bayesian Classification System. in Proceedings of the Fifth International Conference on Machine Learning. University of Michigan, Ann Arbor. : Morgan Kaufman.

Deitterich, T. (1989). Limitations on Inductive Learning. in Proceedings of the Sixth International Workshop on Machine Learning. Cornell University, Ithica NY. (pp. 125-128). San Mateo, CA: Morgan Kaufman

DeJong, J., & Mooney, R. (1986). Explanation-based Learning: An Alternative View. Machine Learning, 1(2), pp. 145-176.

Fodor, J. (1985). The Modularity of Mind. Boston, MA: MIT Press.

Frese, M., & Sabini, J. (1985). Goal Directed Behavior: The Concept of Action in Psychology. Hillsdale, NJ: Lawrence Erlbaum Associates.

Hoffman, C., Mischel, W., & Mazze, K. (1981). The Role of Purpose in the Organization of Information About Behavior: Trait-based Versus Goal-based Categories in Person Cognition. Journal of Personality and Social Psychology, 39, 211-255.

Horvitz, E., Cooper, G., & Heckerman, D. (1989). Reflection and Action Under Scarce Resources: Theoretical Principles and Empirical Study. (Knowledge Systems Laboratory Working Paper No. KSL-89-1). Stanford University, Stanford, CA

Hunter, L. (1989). Knowledge Acquisition Planning: Gaining Expertise Through Experience. PhD. thesis, Yale University, New Haven, CT. (Technical Report YALEU/DCS/TR-678)

Hunter, L. (1990a). Knowledge Acquisition Planning for Inference from Large Datasets. in Proceedings of the 23rd annual Hawaii International Conference on System Sciences, Software Track. Vol. 2 Kona, Hawaii. (pp. 35-44). Washington, DC: IEEE Press.

Hunter, L. (1990b). Deciding What to Learn. Submitted to the Seventh International Conference on Machine Learning. Austin, TX.

Langley, P., Gennari, J., & Iba, W. (1987). Hill Climbing Theories of Learning. Proceedings of the Fourth International Workshop on Machine Learning. Irvine, CA. (pp. 312-323). : Morgan-Kaufman.

Lehnert, W. (1978). The Process of Question Answering. Hillsdale, NJ: Lawrence Erlbaum Associates.

Livesey, P. (1986). Learning and Emotion: A Biological Synthesis. Volume 1, Evolutionary Processes. Hillsdale, NJ: Lawrence Erlbaum Associates.

Ram, A. (1989). Question-driven Understanding: An Integrated Theory of Story Understanding, Memory and Learning. PhD. thesis, Yale University, New Haven, CT. (Tech report YALEU/CSD/RR#710)

Rumelhart, D., McClelland, J., & the PDP Group (1986). Parallel Distributed Processing: Explorations in the Microstructure of Cognition (Volumes 1, 2 & 3). Cambridge, MA: MIT Press.

Sacerdoti, E. (1971). A Structure for Plans and Behavior. New York, NY: American Elsvier.

Schank, R. (1982). Dynamic Memory: A Theory of Reminding and Learning in Computers and People. Hillsdale, NJ: Lawrence Erlbaum Associates.

Schank, R., & Abelson, R. (1977). Scripts, Plans, Goals and Understanding. Hillsdale, NJ: Lawrence Erlbaum Associates

Schank, R., Collins, G., & Hunter, L. (1986). Transcending Inductive Category Formation In Learning. Behavioral and Brain Sciences, 9(4), pp. 639-687.

Srull, T., & Wyer, R. (1986). The Role of Chronic and Temporary Goals in Social Information Processing. in R. Sorrentino, & E. Higgins (eds.), Handbook of Motivation and Cognition: Foundations of Social Behavior. (pp. 503-549). Guilford, CT: The Guilford Press.

Tolman, E. (1932). Purposive Behavior in Animals and Men. New York: Century Press.

Utgoff, P. (1986). Shift of Bias for Inductive Concept Learning. in R. Michalski, J. Carbonell, & T. Mitchell (editors.), Machine Learning 2 Los Altos, CA: Morgan Kaufmann.

Valient, L. (1984). A theory of the learnable. Communications of the ACM, 27(11), 1134-1142.

Zukier, H. (1986). The Paradigmatic and Narrative Modes in Goal-Guided Inference. in R. Sorrentino, & E. Higgins (eds.), Handbook of Motivation and Cognition: Foundations of Social Behavior. (pp. 465-502). Guilford, CT: Guilford Press.

Medline is a trademark of the National Library of Medicine.

Brainstormer: A Model of Advice-Taking*

Eric K. Jones
Institute for the Learning Sciences
Northwestern University

Abstract

Research on advice-taking in artificial intelligence is motivated by the promise of knowledge-based systems that can accept high-level, human-like instruction [11]. Examining the activity of human advice-taking is a way of determining the key computational problems that a fully-automated advice taker must solve.

In this paper, we identify three features of human advice-taking that pose computational problems, and address them in the context of BRAINSTORMER, a planning system that takes advice in the domain of terrorist crisis management.

1 The Advice-Taking Task

Advice is information communicated with the intent of helping an agent make progress on a problem. Taking advice therefore involves understanding how the advice is relevant to solving a problem and acting on this understanding. We have identified three characteristic features of the human advice-taking task, each of which imposes important computational demands on the advice taker. We believe that a fully-automated advice taker has to give an account of each of these features.

1. **Empowerment:** Taking advice empowers an agent to do something by furnishing him or her with information. Consequently, an advice taker must be capable of translating initial, high-level advice into instructions that can be carried out.
2. **Situatedness:** Advice addresses the needs of an agent situated in the context of trying to solve a particular problem. It follows that an advice taker should be able to relate advice to needs for information that have arisen in the course of problem solving.
3. **Communication:** People never communicate everything that is relevant; instead, they provide only information sufficient for a recipient to reconstruct the intended message from context. Advice is communicated information; an advice taker thus has to be able to infer information that is relevant but not explicitly provided.

In the remainder of this section, we describe the computational demands that each of these features of advice-taking place on an advice taker, and discuss the extent to which past research has addressed these demands.

1.1 Advice-taking as empowerment

Most past research on advice-taking has focused on showing how taking advice can empower an agent to do something. Research into this aspect of advice-taking falls into two categories. One category of research has investigated the problem of converting high-level, human-like advice into concrete or *operational* instructions that a machine can carry out [6, 11, 12]. This research advocates that view of an advice taker as a kind of compiler for a very high-level language. Mostow's program FOO [12], for example, translates high-level advice about playing the game of hearts, (*"avoid taking points"*), into a heuristic-search problem solver that can carry out that advice.

*This research was supported in part by the Air Force Office of Scientific Research (AFOSR). The Institute for the Learning Sciences was established in 1989 with the support of Andersen Consulting, part of The Arthur Andersen Worldwide Organization.

A second category of research starts with the observation that that a major difficulty in empowering knowledge-based systems is the so-called knowledge acquisition bottleneck: the problem of getting knowledge into the system sufficient for it to perform interesting tasks. This leads naturally to the view of an advice taker as mechanism for streamlining knowledge acquisition [1, 2, 3, 14]. The TEIRESIAS system [1, 2], for example, guides knowledge acquisition in a way that encourages completeness and consistency with existing knowledge, and also provides high-level tools to facilitate debugging problems that arise from faulty or missing knowledge.

1.2 Advice-taking as a situated activity

Unfortunately, more is involved in taking advice than converting it into an operational vocabulary and ensuring consistency. Once one considers advice-taking as a *situated* activity, further demands are placed on the advice taker. In particular, a situated advice taker must be able to use the advice it receives to address needs for information that arise in the context of current problem solving.

Systems such as FOO and TEIRESIAS do not treat advice-taking as a situated activity. In FOO, for example, there is no current problem solving because no problem solver exists prior to advice-taking. Instead, FOO builds a complete problem solver from scratch during advice-taking, using the input advice as a specification. TEIRESIAS is concerned with acquiring knowledge for unspecified future problem solving, not acquiring knowledge needed to progress on a current problem.

There are systems that direct advice towards current problem-solving, ([5, 10, 16] for example), but they all make assumptions that drastically simplify the interaction between advice giver and advice taker: they assume that advice is supplied immediately after a need for information arises, and that the advice comes with detailed instructions about how it is to be used. For example, the Soar system [5] can use search-control advice in current problem solving, but advice is provided as soon as an operator selection difficulty is encountered, and the system simply presumes that any advice it receives is intended to be used to resolve that particular difficulty.

In a more realistic setting, neither of these simplifying assumptions holds. First, an advice-giver cannot be expected to be on hand the instant a need for information arises. A problem solver that is engaged in a complex problem solving task can be expected to have a large set of current needs, some of which it may succeed eventually in satisfying by itself, others of which may require outside assistance. An advice giver should be able to give advice that relates to any current need, whether or not it is in the current "focus of attention" of the problem solver.

Second, in a real-world setting, an advice giver cannot be expected to give detailed instructions about how advice is to be used. An ability to give such instructions requires that the advice giver know the exact nature of the problem solver's needs, which in turn requires that it have a detailed knowledge of the arcane internal workings of the problem solver. For example, the instructible Soar system described above requires that the user know about operator selection and about the details of particular operators, both of which kinds of knowledge should be hidden from the user.

For these reasons, a fully automated advice taker must be equipped to relate advice to the problem solver's current needs for information. This implies that the advice taker know what these needs are. At the very least, the advice taker should be able to record needs for information as they arise, and store them in such a way that they can be accessed when advice is presented that addresses them.

1.3 Advice-taking as communication

Advice is not just information relevant to a specific problem, it is *communicated* information. An advice taker is guided by the expectation that the advice giver intends the advice to be helpful in current problem solving. For the advice to be helpful, it should address a current need for information. If there is an operational form of the advice that can easily be seen to satisfy a current need, then the advice taker's task is simple. But there will also arise situations where adducing the advice giver's intentions is less straightforward.

People expect discrepancies between the information that advice ostensibly provides and the information that they actually need. These discrepancies can arise in the course of human communication for reasons of communicative efficiency, or because the advice giver has an incorrect or incomplete understanding of the needs of the advice taker. In either case, the advice taker expects to have to bridge the

gap between an initial, "literal" interpretation of the advice and a derived interpretation that makes clear how the advice addresses an actual need for information.

Furthermore, even if a need for information can be determined that the advice plausibly addresses, the advice may not be presented in sufficient detail for it to adequately satisfy the need. In that case, the advice taker should try to fill in the missing details; failing that, it should ask the advice giver for more information.

While some existing advice-taking systems are capable of asking the user to fill in missing details [2], few are capable of inferring these details on their own, (the MOLE system [3] is a noteworthy exception). Furthermore, existing systems cannot bridge the gap between ostensible and actual needs for information. In the remainder of this paper, we introduce the BRAINSTORMER system, and show how considering advice-taking as situated activity that involves communication has informed BRAINSTORMER's design.

2 Brainstormer: A Planner That Takes Advice

Our theory of advice-taking is implemented in BRAINSTORMER, a planning system that takes advice in the domain of terrorist crisis management. BRAINSTORMER translates abstract advice into specific directives about how to make progress on a problem it is working on. Advice is presented in the form of proverbs, represented in an abstract vocabulary of planning-relevant knowledge[7, 13, 15]. BRAINSTORMER uses the advice to come up with plans for countering terrorism.

BRAINSTORMER has two top-level modules: the **planner** and the **adapter**. The planner is BRAINSTORMER's problem solver and is similar in some ways to STRIPS [4]. It suggests plans for countering terrorism. The adapter is BRAINSTORMER's advice taker. It acts as an intelligent assistant to the planner, using advice it receives to address pre-existing needs of the planner for information. In principle, the planner could operate on its own, but in practice it relies heavily on the adapter for assistance.

BRAINSTORMER provides an account of each of the aspects of advice-taking discussed above.

1. **Advice-taking as empowerment.** Like other advice takers, BRAINSTORMER's adapter has rules for translating initial, high-level advice into the operational vocabulary of the planner. Once this advice has been transformed into an operational vocabulary, it is in a form that can directly match the planner's needs for information. If suitable needs can be found, then the advice empowers the planner to make forward progress in planning.

2. **Advice-taking as a situated activity.** BRAINSTORMER is presented with advice during ongoing problem solving. The planner is first handed a planning problem and allowed to run to quiescence. Only then is advice presented. Although the planner come up with a variety of plan suggestions on its own, it will also have discovered needs for information that it is unable to satisfy by itself, and which if satisfied would allow it to make further progress in planning. These needs arise whenever the planner tries to retrieve a knowledge structure and fails. BRAINSTORMER monitors needs for information as they arise, recording them in memory as *queries*.

   Queries are stored in terms of features of potential answers. Once advice has been converted into the operational vocabulary of the planner, features of the advice are used as cues for retrieving queries that the advice might address. Attached to each query is a procedure that specifies how the planner should use answers to it. If a query is retrieved, and if the advice used to retrieve it constitutes an adequate answer, then the query's procedure is invoked. Planning can then continue at the point in the planning process where the need for information originally arose.

3. **Advice-taking as communication.** Taking advice in BRAINSTORMER, then, is defined as turning a proverb into an adequate answer to a pre-existing query. Proverbs are a very minimalist form of communication in which all details of the specific problem that they are intended to address are omitted. BRAINSTORMER thus expects input advice to be sketchy and incomplete. After translating advice into an operational vocabulary, there may still be no query

PROBLEM SITUATION: *A terrorist crisis involving terrorists from the P.L.O., in which five American hostages were taken and one was killed.*

GOALS: *Prevent future terrorist crises similar to the problem situation,* and *take action against the agents responsible for the terrorist crisis.*

Figure 1: An example planning problem.

1. *Find a plan for the goal of preventing future hostage holdings similar to the one in the problem situation*
2. *Find a plan for the goal of preventing future killings similar to the one in the problem situation*
3. *Find a plan for the goal of preventing future terrorist crises similar to the one in the problem situation*
4. *Find an explanation for the hostage holding*
5. *Find an explanation for the killing*
6. *Find an explanation for the terrorist crisis*

Figure 2: Some queries posted during planning.

in memory that directly matches it, or if there is, the advice may supply insufficient information to serve as an adequate answer.

It follows that BRAINSTORMER must confront the following problems: (1) finding queries that advice addresses and (2) turning advice into adequate answers. The adapter is equipped with two sets of inference strategies for coping with these problems. These strategies add elements to the advice that were not explicitly present in the input but which the advice giver may have intended to communicate implicitly. The first set of strategies allows the adapter to hypothesize additional components of the advice sufficient to retrieve a query that the advice addresses. Once a query is found, the second set of strategies allows the adapter to fill in missing details needed for the advice to provide an adequate answer to the query. In carrying out these kind of inferences, the adapter operates on the assumption that the user intended the input advice to be helpful but omitted certain details for the usual reasons that information is often left implicit in ordinary communication.

3 An Example Planning Problem and Initial Planning

In this section, we sketch a typical planning problem in BRAINSTORMER and describe some of the queries that arise from planning for this problem, prior to the presentation of advice. Planning problems are represented using **problem situations** and **goals**. A natural-language paraphrase of a representative planning problem is shown in figure 1, above. The plans that BRAINSTORMER comes up with are plans for goals in this problem.

BRAINSTORMER's planner is first invoked on this planning problem without the help of any advice and permitted to run to quiescence. In the process of planning, numerous queries are posted, each of which signals a potential opportunity for further planning. Some of these queries are listed in figure 2, above. Queries 1, 2, and 3 were posted when the planner failed to directly retrieve a plan for a goal. Queries 4, 5, and 6 request knowledge structures that can match preconditions of an abstract counter-planning rule. Given an explanation for the occurrence of an undesirable event, this rule can transform a goal to prevent the recurrence of a similar event into a goal to prevent the recurrence of the conditions that led to it[1]. Queries 4, 5 and 6 were posted on separate occasions where the planner tried to run this rule and failed to retrieve suitable explanations.

When the planner can make no further progress on its own, a user presents BRAINSTORMER with advice in the form of a proverb. BRAINSTORMER's

[1] In the context of goals to prevent future occurrences of similar events, an event is considered similar to a second event if there is an explanation for the first, a non-vacuous generalization of which is also an explanation for the second.

INPUT:	• The planning problem and queries described above. • The proverb *he who has suffered more than is fitting will do more than is lawful.*
OUTPUT:	• (Elaboration of the problem situation) *The terrorists are Palestinian, and were suffering because of poor living conditions in refugee camps.* • (Plan suggestion) *Build public housing projects for the refugees.*

Figure 3: An example that illustrates the problem of producing adequate answers

adapter then has the task of turning this proverb into an adequate answer to a query. In the previous section, we noted that in carrying out this task, the adapter must confront the problems of finding a query that the advice addresses and of fleshing out the advice into an adequate answer to it. As intimated above, these problems follow directly from the demands imposed by considering advice-taking as a situated activity involving communication. In the next two sections, we sketch BRAINSTORMER's approach to dealing with these problems.

4 Producing Adequate Answers

We start with the problem of adequate answers. Once a query has been found that advice addresses, BRAINSTORMER's adapter must fill in any details needed for the advice to constitute an adequately specific and plausible answer to the query. An apt illustration of this kind of processing arises during BRAINSTORMER's processing of the proverb *"he who has suffered more than is fitting will do more than is lawful."* BRAINSTORMER's input-output behavior is summarized in figure 3.

BRAINSTORMER came up with this output by using the proverb to answer query 6 above, (*"find an explanation for the terrorist crisis"*). One way that the proverb cen be represented in the operational vocabulary of the planner is as *an explanation for the occurrence of an illegal action: the illegal action happened because some agent was suffering, and that explains why the agent carried out the illegal action.* This interpretation directly matches queries 4, 5, and 6 above: hostage-holdings, killings, and terrorist crises can all be viewed as illegal actions, and the proverb can therefore provide an explanation for each. BRAINSTORMER prefers queries whose answers subsume answers to other queries; in this case, an explanation for the terrorist crisis subsumes explanations for the killing and hostage holdings, because each of these actions was part of the terrorist crisis. Consequently, BRAINSTORMER assumes that the advice is best treated as answering query 6[2].

Although the adapter possesses an operational representation of the advice and a query that it plausibly addresses, its task is not over. The advice is still missing important details which it is the adapter's responsibility to fill in. Once the adapter has committed to answering query 6, the representation of the advice can be paraphrased as follows: *the terrorists were suffering, and that explains why they carried out the terrorist crisis.* As it stands, this representation constitutes a hypothesis, that the terrorists were suffering, and that this explains why they carried out the terrorist crisis. For BRAINSTORMER to accept this hypothesis as an adequate answer to query 6, it requires that any unsupported components of the hypothesis be justified.

The adapter has a small set of strategies for justifying unsupported components of hypotheses. These include the following.

- **Memory search:** Use an unsupported component of a hypothesis as an index into memory for retrieving a knowledge structure that supports it.
- **Communicative assertion:** If only one unsupported component of a hypothesis remains, assume that the user intended the advice to assert that this component in fact holds.
- **Ask the user.**

Asking the user is hardly a novel idea: a variety of existing systems treat advice-taking as mixed-initiative knowledge acquisition, in which the system directs a user to fill in all missing details. In BRAIN-

[2]Our discussion here omits many important issues, including how the proverb is initially represented, how it is transformed into the operational vocabulary of the planner, and how the operational interpretation is used to retrieve relevant queries. For a discussion of these and other issues, see [7, 8, 9].

INPUT:	• The planning problem and queries described in section 2. • The proverb *an old poacher makes the best keeper.*
OUTPUT:	• (Plan suggestion) *Hire a former terrorist to defend against terrorism*

Figure 4: An example that illustrates the problem of finding relevant queries.

PRECONDITIONS:	**plan:** *a plan of defense* **parameter:** *the actor of the plan* **situation:** *the plan of defense is part of a goal conflict involving stereotyped attackers and defenders*
POSTCONDITION:	*A new plan whose actor satisfies the attacker's stereotype*

Figure 5: The proverb *"an old poacher makes the best keeper"* represented as a plan refinement operator.

STORMER however, asking the user is a strategy of last resort. Wherever possible, the adapter attempts to arrive at an adequate plausible interpretation of the advice on its own.

In the current example, there are two unsupported hypothesis components: that *the terrorists were suffering*, and that *this suffering explains why they carried out the terrorist crisis*. The adapter employs its **memory search** strategy to justify the first. Specifically, it uses the fact that the terrorists are members of the P.L.O. to retrieve a standard explanation for Palestinian suffering: poor living conditions in the refugee camps. The **communicative assertion** strategy is then sufficient to justify the second component.

Once the adapter has come up with what it judges to be an adequate answer to a query, the planner is reinvoked at the point in its processing that the query arose. Recall that this query was posted when the planner was trying to satisfy a precondition of a general counterplanning rule. The rule uses explanations of the occurrence of an undesirable event to transform a goal to prevent the recurrence of a similar event into goals to prevent the recurrence of the conditions that led to it. In the current instance, the undesirable event was the terrorist crisis. Failure to retrieve a suitable explanation from memory led the planner to post the query. The proverb provides the missing explanation: the terrorist crisis happened because of poor living conditions in the refugee camps. Now that it has this explanation, the planner can apply the counterplanning rule. This leads the planner to establish a new goal of improving living conditions in the refugee camps and eventually, to propose building public housing projects.

5 Finding Relevant Queries

In the preceding example, BRAINSTORMER encountered little difficulty trying to find a relevant query, because the operational form of the advice, (an explanation), directly matched a pre-existing query. In other cases however, this lucky state of affairs does not obtain. For example, suppose BRAINSTORMER is handed the proverb *"an old poacher makes the best keeper"*. BRAINSTORMER's input-output behavior is summarized in figure 4.

This time, the relevant operational form of the proverb is a *plan refinement operator*, which is a rule that the planner can use to elaborate plans it proposes. The representation of this operator is paraphrased in figure 5. This operator does not directly match any existing queries. If a suitable query already existed, the planner would have to have already considered a plan of defense for one of its goals, and be looking for ways to refine it. This however is not the case in the current example. Consequently, further work is necessary to find the query that the advice addresses.

The following is a highly informal sketch of the reasoning that the adapter goes through in searching for a relevant query. (For a more technical discussion, see [8]).

- The user has suggested a way of refining a plan of defense. She is giving me advice, so she is trying to help me solve my current problem, namely preventing future terrorist crisis similar to the one in the problem situation, (see figure 1). Thus, she believes that this plan refinement operator supplies information that is useful for this purpose. For this operator to be useful, a

plan of defense would also have to be useful. But I haven't considered any plans of defense. Maybe I should have considered a plan of defense for one of my goals.

- One of my goals is preventing future terrorist crises similar to the one in the problem situation. Could a plan of defense be useful in planning for this goal? Yes, it could. A plan of defense can be effective at preventing future occurrences of the kinds of events it defends against, because it raises their expected cost.
- OK, I'll *hypothesize* that a plan of this sort is actually useful for this goal in the current context. After all, maybe the user intended to communicate this hypothesis. That would explain why she believes that I should find this operator useful. Now I know how to use the operator.

In carrying out the above reasoning, the adapter locates a query that the advice addresses by elaborating the representation of the advice to include a hypothesis that a plan of defense is appropriate for the goal of preventing future terrorist crises. Next, control returns to the planner, which goes on to apply the plan refinement operator to this hypothesized plan and eventually suggests *hire a former terrorist to defend against terrorism.*

There are two important morals to draw from this example. First, the inference that the defense plan might be appropriate was *licensed by* the assumption that the user was trying to be helpful in the context of the given problem. This assumption both legitimates the final interpretation of the advice and justifies the inferential effort expended in finding it. An advice-taking system that does not reflect the idea that taking advice involves making sense of *information communicated with the intention of being helpful* is unable to make these inferences; consequently, in this situation at least, such an advice-taking system would be unable to profit from the advice.

The second moral of this example is that an advice taker needs knowledge about the problem solver's problem solving *process* in order to carry out this kind of inference. The requisite knowledge is over and above information about the advice-taker's dynamic needs for information. In the above example, the adapter had to be able to reason about how the planner *uses* plan refinement operators: it uses them when it has already considered a plan of defense for one of its goals. Carrying out this reasoning requires that the adapter know that the planner's planning process takes goals as input, suggests plans for those goals, and then suggests refinements to those plans. Previous advice-taking systems lack this kind of self-knowledge, and are consequently unable to perform this kind of reasoning.

6 Conclusion

We began this paper by identifying three characteristic features of human advice taking that should inform a computational theory of advice-taking. Advice *empowers* an advice taker, advice-taking is *situated* in the context of ongoing problem solving, and advice-taking involves *communication*.

BRAINSTORMER is an advice-taking system that takes account of these features of human advice-taking. First, advice helps BRAINSTORMER's planner to come up with plans for countering terrorism. Second, the planner's needs for information are recorded during planning as they arise, allowing BRAINSTORMER to use advice to facilitate ongoing problem solving. Third, BRAINSTORMER is equipped with inference strategies that allow it to infer some of the kinds of details that are commonly left implicit in ordinary human discourse. In summary, BRAINSTORMER represents a more comprehensive model of human advice-taking than earlier advice-taking systems, and constitutes a step in the direction of problem solvers that can accept high-level, human like instruction.

Acknowledgements

Thanks to Eric Domeshek for comments on a draft of this paper, and to Andy Fano for help with the program.

References

[1] Randall Davis. Knowledge acquisition in rule-based systems — knowledge about representation as a basis for system construction and maintenance. In D.A. Waterman and F. Hayes-Roth, editors, *Pattern Directed Inference Systems*. Academic Press, New York, 1978.

[2] Randall Davis. TEIRESIAS: Applications of meta-level knowlege. In Randall Davis and D.B. Lenat, editors, *Knowledge-Based Systems in Artificial Intelligence*, pages 229–484. McGraw-Hill, New York, 1982.

[3] Larry Eshelman and J. McDermott. MOLE: A knowledge acquisition tool that uses its head. In *Proceedings AAAI-86 Fifth National Conference on Artificial Intelligence*, pages 950–955, Philadelphia, PA., August 1986. AAAI.

[4] Richard E. Fikes and N.J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2:189–208, 1971.

[5] Andrew Golding, P.S. Rosenbloom, and J.E. Laird. Learning general search control from outside guidance. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, Milan, Italy, August 1987. IJCAI.

[6] Frederick Hayes-Roth, P. Klahr, and D.J. Mostow. Advice-taking and knowledge refinement: An iterative view of skill acquisition. In John R. Anderson, editor, *Cognitive Skills and Their Acquisition*, pages 231–253. Lawrence Erlbaum Associates, Hillsdale, N.J., 1981.

[7] Eric K. Jones. Case-based analogical reasoning using proverbs. In Kristian Hammond, editor, *Proceedings: Case-Based Reasoning Workshop*, Pensacola Beach, FL., May 1989. Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc.

[8] Eric K. Jones. Learning by taking advice: The need for a model of the problem-solving process. Paper submitted to the Seventh International Conference on Machine Learning, 1990.

[9] Eric K. Jones. BRAINSTORMER: A situated advice taker. Paper submitted to the Eighth National Conference on Artificial Intelligence, 1990.

[10] Richard L. Lewis, A. Newell, and T.A. Polk. Towards a SOAR theory of taking instructions for immediate reasoning tasks. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, pages 514–521, Ann Arbor, Michigan, August 1989. Cognitive Science Society.

[11] John McCarthy. Programs with common sense. In Marvin Minsky, editor, *Semantic Information Processing*. MIT Press, Cambridge, MA., 1968.

[12] David J. Mostow. Machine transformation of advice into a heuristic search procedure. In Ryszard S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, pages 367–404. Tioga Publishing Company, Cambridge, MA., 1983.

[13] Christopher Owens. Domain-independent prototype cases for planning. In Janet Kolodner, editor, *Proceedings of a Workshop on Case-Based Reasoning*, pages 302–311, Clearwater, FL., May 1988. Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc.

[14] Michael D. Rychener. The instructible production system: A retrospective analysis. In Ryszard S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, pages 429–460. Tioga Publishing Company, Cambridge, Mass, 1983.

[15] Roger C. Schank. *Explanation Patterns: Understanding Mechanically and Creatively*. Lawrence Erlbaum Associates, Hillsdale, NJ., 1986.

[16] Donald A. Waterman. Generalization learning techniques for automating the learning of heuristics. *Artificial Intelligence*, 1:121–170, 1970.

Representing abstract plan failures

Christopher Owens
The University of Chicago

An intelligent agent must be able to recover from and learn from its failures. This involves building a causal explanation of why the failure occurred and using that causal explanation as the basis of further reasoning about how to deal with the failure. This paper argues that the tasks of building the explanation and reasoning from the explanation should be tightly coupled, to avoid the problem of factually correct but pragmatically useless explanations. This integration can be accomplished by using a model of reasoning about plan failures that is based upon knowledge structures that link descriptions of stereotypical plan failures with descriptions of repair and recovery strategies appropriate to those failures.

1 Dealing with failures

An intelligent agent operating in an uncertain environment must constantly face the fact that its plans may fail. Knowledge about the world is incomplete, knowledge about the necessary preconditions for a particular course of action is likely to be underspecified, and the state of the world might change between the time a particular action is planned and the time it is executed.

While some failures are the result of circumstances that are genuinely beyond the control and predictive abilities of the agent, others result from planning errors which can be corrected if the agent understands the connection between the planning error and the failure. A widely-accepted paradigm for dealing with failed plans (see, for example, [Sussman, 75], [Hayes-Roth, 83], [Hammond, 86], [Birnbaum and Collins, 88]), is that an agent must:

- **Explain** the failure. Assign blame for the failure to some condition over which the agent could have had control. (Or, if no such condition can be found, identify the failure as an unforseeable, unavoidable one.)
- **Recover** from the failure. Plan some activity that will lead toward the original goals, taking into account the changed world resulting from the failure. Or, if achieving the original goal now looks too expensive, work on some other goal.

- **Repair** the plan that resulted in the failure. If the explanation assigns blame to some condition internal to the planner, for example failure to look for a certain contraindicating condition before commencing a particular activity, modify the plan so that future uses of the same plan will not result in the same failure.

It is clear that under this model, the steps of recovering from a failure and repairing the plan depend upon a good explanation – one that assigns blame to some state of the world or state of the planner's knowledge that is both responsible for the failure and within the power of the agent to change. It is also true, though less immediately clear, that whatever process builds the explanation of the plan failure must be informed by the system's knowledge of recovery and repair.

To understand this latter point, consider the example of a robot that, given the instruction to carry two chairs from one room to the next, picks up both chairs together and, in the process of trying to maneuver through a narrow doorway, damages the chairs. Some potential explanations of the failure are:

1. The doorway was too narrow.
2. The chairs were too large.
3. The robot is poor at judging the widths of passageways.
4. The robot did not know to re-estimate its size when carrying loads.
5. The robot knew to re-estimate its size, but is poor at it.
6. The robot did not know about the vulnerability of chairs to impact damage.
7. The robot chose the plan of carrying two chairs at once even though the plan was risky, because it considered the decrease in time worth the increased risk of damage (This might not be an error, but might in fact be a reasonable course of action if, for example, the room were on fire.)

Although each of these explanations refers to the same manifestation of a failure (i.e. dented chairs), the repairs and recoveries derived from each will be different. The first two explanations assign blame to states of the world that aren't reasonably within the robot's control. The recoveries or repairs they suggest (make the doorway larger, for example) are not feasible. The next four explanations deal with errors or omissions in the robot's knowledge about carrying objects, and the repairs they suggest involve modifying or adding to that knowledge. The final explanation deals with the mechanism the robot uses to mediate between competing goals, and it suggests modifying that mechanism.

Examining the difference between the first two explanations and the others points out that the goodness of an explanation depends not only upon the degree with which it faithfully captures some aspect of the causality leading up to the failure, but also upon the degree to which it suggests an operational repair. This dependency has implications for the role of memory in reasoning about failures.

2 Building explanations

There are basically two mechanisms whereby a system can build a useful explanation of a failed plan: "Build from scratch", which involves chaining together causal rules into an explanation that connects the unexpected failure with known conditions, and "Retrieve and apply", whereby the system maintains a library of abstract explanations that can be matched against the circumstances of a failure and instantiated with the particulars of the situation.

The "retrieve and apply" method of explanation construction is discussed in [Schank, 86] and [Kass *et al.*, 86]. The fundamental idea is that certain fairly complex patterns of causality tend to recur in the course of an agent's interaction with the world. These are not in principle different than the patterns that the system could recognize via the "build from scratch" approach, but the number of them that the system actually encounters is small relative to the total number of syntactically valid patterns that the system could build by chaining together its primitive causal rules. Caching these large and frequently-encountered causal patterns is not only useful from the point of view of efficiency, but also, as has been argued in [Kass and Owens, 88], from the point of view of likely utility in the face of incomplete knowledge.

In the context of reasoning about plan failures, there is a third advantage to the "retrieve-and-apply" approach to explanation, and this advantage pertains to the above-discussed question of how the system's notion of repair and recovery should guide its choices when it builds an explanation of a failed plan. The remainder of this paper deals with the representational form and content of a class of knowledge structure that addresses the task of understanding, recovering from and repairing failed plans.

3 What to represent

As has been argued in more detail in [Schank, 86] and [Owens, 88], there is an interesting correspondence between the type of stereotypical plan failure that an intelligent agent needs to represent, and common advice-giving proverbs and aphorisms. *The lazy man's load*, for example, refers to the failure resulting from trying to minimize the number of trips required to move some number of objects by moving all of them at one time. A more generally applicable pattern describing the same class of error might be the *too many irons in the fire* pattern, which deals with the general problem resulting from overdoing the optimization of combining tasks and executing them simultaneously.

These proverbs categorize situations in some useful way. All *too m y irons* situations share certain causal properties, for example that the failure is related to the fact that multiple tasks are being combined. It is generally easy for people to decide based on a *post facto* description, whether or not a particular situation is of the *too many irons in the fire* type. There is a common set of failure recovery strategies that should be successful in all *too many irons* situations, for example interleaving fewer tasks, putting some tasks on hold or finding someone to help. People are able to enumerate these strategies when asked about *too many irons* situations in the abstract. The goal for knowledge structures to be used as

part of an intelligent planning system, is to similarly categorize experiences into meaningful abstract classes or clusters.[1]

It is the observation that proverbs seem to link abstract failure descriptions with abstract recovery strategies that addresses the question of how the tasks of explaining, recovering from and repairing failures can be better integrated. They are done so via a knowledge structure that combines an abstract description of a plan failure with recovery and repair strategies. It is called a **plan failure explanation pattern**, or PFXP, and it is an extension and specialization of the **XP**s described in [Schank, 86] and [Kass *et al.*, 86].[2]

3.1 Understanding the failure

The central content of a PFXP is a template that can be used to build a causal model of the class of situation represented by that knowledge structure. The template characterizes, in abstract terms, the common causality that all situations covered by this PFXP share. In the case of the *Too many irons in the fire* situation, the causal pattern that characterizes the failure, when written out in English, looks something like:

> Simultaneously executing multiple tasks can fail if several of the tasks have time-critical steps, in that the agent will be busy attending to one time-critical step at the time that another time-critical step requires the agent's attention, or if the total workload of the tasks exceeds the agent's capabilities, in that the agent will be overloaded.

By deciding that this particular causal pattern applies to the current situation, the system makes several commitments about which features of the current situation will participate in the explanation. The fact that the object being moved is a chair, for example, is not involved in matching the situation against this pattern, and is consequently ignored, while the fact that the situation involved scaling the size of a task is highlighted. When it comes time to repair the problem, the system can ignore the former and focus on the latter.

3.2 Recovering from the failure

A recovery strategy, such as the decision to restart the plan using less interleaving of tasks, could be calculated dynamically from the causal model. Since the causal model shows that the interleaving is implicated in the plan failure, reformulating the plan to use less interleaving is not a particularly complicated inference. Other recovery strategies, like deciding to look for help, are not quite so easily and cheaply inferred from the initial causal explanation of the plan failure.

[1] See also [Lehnert, 81], [Schank, 82], [Dyer, 82] for a more general discussion of the relationship of proverbs to abstract thematic structures in the context of story understanding.

[2] See also the planning **TOP**s discussed in [Hammond, 86] for another abstract characterization of planning failures

In a system based on PFXPs, though, there is very little advantage to dynamically calculating repair strategies by inferring them from the causal model. Since the causal model is a static object stored as part of the knowledge structure, a set of potential repair strategies can also be packaged with a PFXP in memory. The system's task can therefore be to choose from among several possible recovery strategies rather than to try to build one from scratch. By comparison, in a system that builds explanations from scratch by chaining primitives together there is no obvious place to put recovery strategies in memory.

While the abstract strategies stored with a PFXP are static, the instantiation must of course be handled dynamically, since the abstract knowledge structure cannot know in advance the details of the situations to which it will be applied. The variables that are instantiated in the process of matching the causal model of the PFXP to the current situation, are carried over and used to instantiate the recovery strategy into a specific course of action.

The two recovery strategies mentioned above are representative of two distinct general classes of strategies. The first, re-executing with less interleaving, is basically an application of a general strategy that could be applied to any failure:

> *If some fact* **x** *is causally implicated in the failure, then cause* **x** *not to hold and try the plan again.*

While this strategy is general and simple, it is fraught with problems. A large number of conditions are causally implicated in the failure; yet many of them cannot or should not be changed by the planner, either because they are present in service of some goal or because they are beyond the control of the planner. This recovery strategy, given some of the explanations from the beginning of this paper, could easily generate the "solution" of making the doorway larger.

The second recovery strategy, looking for help, is much more specific to the *too many irons* failure. As a design principle, PFXPs should be set up with this more specific type of recovery strategy whenever possible. There is less inferential complexity involved in instantiating the strategy in the context of the failure, and because the inference chain is shorter, we can have more confidence that the strategy will be appropriate.

3.3 Identifying the bad planning decision

The role of a PFXP in allowing a system to correct its inappropriate planning behavior is that it provides a pointer to a planning decision believed likely to be the one responsible for the failure characterized by this particular PFXP. Often the decision pointed to will be a plan transformation. In the case of the *Too many irons* PFXP, the bad decision suggested looks something like:

> *Enhance this plan by simultaneously performing similar steps*

The type of bad decision that is pointed to by a PFXP can take several forms, among them:

Spurious action Some action was taken that caused the plan to fail; had it not been taken the plan would have succeeded. This is the type of failure that the *too many irons* PFXP embodies.

Omitted action Some action could have been taken to prevent this particular failure, but was not taken. This is the type of failure implicated in, for example, the PFXP corresponding to the proverb *A stitch in time saves nine*, which typifies failures in which some low-cost preventive measure would have avoided a high-cost outcome.

Inappropriate plan choice The choice of plan to accomplish some particular goal or subgoal was inappropriate. This type of bad decision can be referred to by PFXPs such as the one corresponding to the proverb *You can catch more flies with honey than you can with vinegar.*

Inappropriate resource choice The choice of resource with which to implement some plan was inappropriate. An example of this can be found in a PFXP corresponding to some interpretations of the proverb *You can't make a silk purse out of a sow's ear* or *A handsaw is a good thing, but not to shave with* – each of which warns against applying a resource to some task for which it is manifestly unsuited.

Ignored factor Some factor was not taken into account that, if it had been taken into account, would have avoided the failure. An extremely general instance of this failure is found in the *Look before you leap* PFXP.

Spurious factor Some factor was considered that, if it had been ignored, would have avoided the failure. The system paid too much attention to an insignificant or irrelevant factor. A number of decisions related to risk-taking fall into this category, such as those implicated in the proverb *He who waits for a fair wind misses many a voyage*

Inappropriate goal prioritization Some goal was given either excess or insufficient consideration, relative to the other goals on which the system was working at the time. This corresponds to the final explanation for the damaged chairs suggested at the beginning of this paper.

Incorrect assumption From the point of view of credit and blame attribution, this type of bad decision is different from the others in that it isn't really a decision that is pointed to here – the system didn't do anything wrong that led it into the failure, but rather it relied on information that is not correct. The **bad decision** field of this type of PFXP points to the offending assumption.

Some of these descriptions are oversimplified here. For example, **spurious factor**, **ignored factor**, **spurious action** and **omitted action** need not be boolean descriptors as suggested above. In fact, this type of error is much more often a matter of degree than it is of absolutes. The description of a bad decision is less likely to be something like

You paid attention to something you should have ignored

than it is to be something like

You paid more attention to this factor than you should have.

The basic idea behind encoding a bad decision as part of the representation of a PFXP is analogous to the credit assignment function of the basic causal model of the PFXP. Just as the basic causal model attempts to focus credit assignment on aspects of the world that are potentially under the planner's control, the **bad decision** field provides a means to focus internal credit assignment on aspects of the system's planning behavior that are modifiable.

A problem with this kind of credit assignment is that there is no guarantee of accuracy. The particular plan transformation or other decision pointed to is neither necessary nor sufficient as an explanation of the failure. Clearly the decision to interleave tasks does not always result in a bad outcome. Generally, in fact, it is a desirable thing to do. Furthermore, and more importantly from the point of view of constructing explanations, not all instances of the *Too many irons* situation result from an explicit decision to interleave tasks. Sometimes the situation might result from, for example, failing to notice a problem developing as more and more situations requiring immediate attention arise. Or, the situation might result from the conjunction of several seemingly unrelated planning decisions. The contents of the **bad decision** field are of heuristic value rather than of provable correctness.

3.4 Modifying the bad planning decision

Just as each description of a failure in the world has associated with it a recovery strategy, likewise each description of a bad planning decision has associated with it a repair strategy. The role of a PFXP in repair is, however, less clear-cut than it is in the area of developing a causal explanation or recovering from specific failures. This is partially due to the fact that internal credit attribution is not so clear-cut as external credit assignment. In the case of the robot moving the chairs, what, exactly, is the bad decision that resulted in the damage? Was it the decision to carry two chairs at once? To move quickly through the door? Or was it the result of placing too much importance on getting the job done quickly, or not enough importance on preserving the chairs.

If, a system constantly encounters errors of the same type, say *too many irons*, it should be less willing to perform the plan transformation that says to enhance a plan by allocating multiple agents to work on it. If, on the other hand, this type of error never occurs, that is a sign that the system is probably missing opportunities to enhance plans by being too conservative – it should be less fussy about applying that particular plan transformation. Ideally, the way to make the system more fussy about a particular plan transformation is to add preconditions for that transformation, and the way to make it less fussy is to drop preconditions. A very crude approximation to this behavior that requires much less knowledge on the part of the system, is to adjust some numerical parameter corresponding to the likelihood of a particular plan transformation being applied based upon the number of errors encountered that point to that transformation. While the latter approach is easier to implement, it does not allow the kind of learning enabled by the former.

4 Conclusions

Reasoning about failed plans by using an approach based upon matching large causal structures is more than just an efficiency issue related to the compilation of rules into larger rules with more detailed applicability conditions. The approach addresses the question of how to more closely link the task of understanding a plan failure with the task of acting based upon that understanding. Because the only knowledge structures available for matching are those that are linked to operational repair and recovery strategies, the system avoids the problem of generating factually correct but pragmatically useless explanations.

References

[Birnbaum and Collins, 88] L. Birnbaum and G. Collins. The transfer of experience across planning domains through the acquisition of abstract strategies. In J. Kolodner, editor, *Proceedings of a workshop on Case-based Reasoning*, pages 61–79, Palo Alto, 1988. Defense Advanced Research Projects Agency, Morgan Kauffmann.

[Dyer, 82] M. Dyer. In-depth understanding: A computer model of integrated processing for narrative comprehension. Technical Report 219, Yale University Department of Computer Science, May 1982.

[Hammond, 86] K. Hammond. *Case-based Planning: An Integrated Theory of Planning, Learning and Memory*. PhD thesis, Yale University, 1986. Technical Report 488.

[Hayes-Roth, 83] F. Hayes-Roth. Using proofs and refutations to learn from experience. In R. Michalski, J. Carbonell, and T. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, pages 221–240. Tioga, Palo Alto, 1983.

[Kass and Owens, 88] A. Kass and C. Owens. Learning new explanations by incremental adaptation. In *Proceedings of the 1988 AAAI Spring Symposium on Explanation-Based Learning*. AAAI, 1988.

[Kass *et al.*, 86] A. M. Kass, D. B. Leake, and C. C. Owens. SWALE: A program that explains. In *Explanation Patterns: Understanding Mechanically and Creatively*, pages 232–254. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[Lehnert, 81] W. Lehnert. Plot units and narrative summarization. *Cognitive Science*, 5:293–331, 1981.

[Owens, 88] C. Owens. Domain-independent prototype cases for planning. In J. Kolodner, editor, *Proceedings of a Workshop on Case-Based Reasoning*, pages 302–311, Palo Alto, 1988. Defense Advanced Research Projects Agency, Morgan Kaufmann, Inc.

[Schank, 82] R. Schank. *Dynamic Memory: A Theory of Learning in Computers and People*. Cambridge University Press, 1982.

[Schank, 86] R. Schank. *Explanation Patterns: Understanding Mechanically and Creatively*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[Sussman, 75] G. Sussman. *A computer model of skill acquisition*, volume 1 of *Artificial Intelligence Series*. American Elsevier, New York, 1975.

Reasoning With Function Symbols In A Connectionist System

Venkat Ajjanagadde*
Department of Computer and Information Science
University of Pennsylvania

Abstract

One important problem to be addressed in realizing connectionist reasoning systems is that of dynamically creating more complex structured objects out of simpler ones during the reasoning process. In rule based reasoning, this problem typically manifests as the problem of dynamically binding objects to the arguments of a relation. In [1,7], we described a rule-based reasoning system based on the idea of using *synchronous activation* to represent bindings. As done in almost all other connectionist reasoning systems developed so far, there, we restricted our focus on the problem of binding only *static objects* to arguments. This paper describes how the synchronous activation approach can be extended to bind *dynamically created objects* to arguments, to form more complex dynamic objects. This extension allows the rule-based reasoning system to deal with function symbols. A forward reasoning system incorporating function terms is described in some detail. A backward reasoning system with similar capabilities is briefly sketched and the way of encoding long-term facts involving function terms is indicated. Several extensions to the work are briefly described, one of them being that of combining the rule based reasoner with a parallelly operating equality reasoner. The equality reasoner derives new facts by substituting equivalent terms for the terms occurring in the facts derived by the rule-based reasoner.

1 Introduction

Suppose, someone told us "At around 7PM last Thursday, as John was walking on Chestnut street,...". Virtually immediately after hearing this very small piece of explicitly stated information, we would have performed a lot of inferences, say, for example, that "John was awake", "He was moving", "His eyes were open", "His feet were touching the ground", "It was probably quite dark " and so on... . In doing this, we would have applied our general knowledge about 'walking', 'light conditions at around 7PM' etc. to make inferences about the particular case of John's walking. One of the well explored approaches of modelling this inferencing phenomenon is the following: Represent the general knowledge using rules such as[1],
walk-on(x,y)$\Rightarrow awake(x)$, $awake(x) \Rightarrow$ *open(eye-of(x))*, *walk-on(x,y)* $\Rightarrow$*touch(feet-of(x),y)* etc.. Now, given the fact *walk-on(john,chestnut-st)*, binding *john* and *chestnut-st* to appropriate variables in the rules, facts such as *awake(john)*, *touch(feet-of(john),chestnut-st)* can be derived.

We believe that this attempt to model the human inference process as reasoning with rules and facts is *a step* in the right direction. But, at the same time, we also find compelling reasons to adopt a connectionist framework in further explorations of this approach. One primary reason is that of inferencing speed. To characterize the knowledge underlying human cognition, a very very large number of rules of the form mentioned above will be

*This work was partially supported by NSF grants IRI 88-05465, MCS-8219196-CER, MCS-83-05211, DARPA grants N00014-85-K-0018 and N00014-85-K-0807, and ARO grant ARO-DAA29-84-9-0027. I am indebted to Lokendra Shastri, whose ideas and suggestions have greatly contributed to this work. John Barnden provided useful comments; my thanks to him.

[1]When not mentioned otherwise, the variables occurring in a rule are assumed to be universally quantified and have scope over the whole rule.

needed. In spite of possessing that vast a knowledge base, human beings perform a broad class of inferences involved in cognition extremely fast, within a few hundred milliseconds. To be acceptable, any computational account of cognition should be able to explain this remarkable phenomenon. One necessary (but not sufficient) condition for achieving such an efficiency is massive parallelism at the knowledge level - the kind of parallelism that is characteristic of connectionist models. Another important reason that suggests the use of connectionist approach is the brittleness of logical rules. Inherent to the connectionist methodology are elegant mechanisms for avoiding this brittleness in a natural fashion.

As a preliminary step towards the realization of a connectionist soft reasoning system, we have addressed the problem of how to reason with hard logical rules in a connectionist system[2]. That meant devising a scheme for encoding rules and facts as a connectionist network and finding a way of realizing the inference process as spread of activation in the network, there being no central controller. Unfortunately, in doing this, many challenging problems surface, which need to be solved.

One of the major problems to be addressed is "How do we represent structured objects *dynamically*[3], i.e., as temporary patterns of activity generated during network's operation?". Let us look at this problem more closely with an example of reasoning.

Consider the rules $fly(x,y,z) \Rightarrow move(x,y,z)$ and $move(x,y,z) \Rightarrow reach(x,z)$. Now, starting from the fact $fly(tweety, tree1, tree2)$ (i.e., $tweety$ flew from $tree1$ to $tree2$), using the first rule we can infer $move(tweety, tree1, tree2)$. From this newly derived fact, using the second rule, we can, in turn, infer $reach(tweety, tree2)$. In a connectionist network performing this reasoning, activity patterns representing the facts $move(tweety, tree1, tree2)$ and $reach(tweety, tree2)$ will have to be generated during the reasoning process.

Let us look at the problem of representing the dynamic objects $move(tweety, tree1, tree2)$, and $reach(tweey, tree2)$ more closely. These are structured objects formed by binding some objects to the arguments of relations. For example, $move(tweety, tree1, tree2)$ is formed by binding the objects $tweety$, $tree1$, and $tree2$ respectively to the first, second, and third arguments of the relation $move$. Thus, the problem of representing these facts boils down to the connectionist variable binding problem[3,8]: How do we represent the binding of an object to an argument of a relation?

In [1,7], we described what we refer to as the *synchronous activation approach* to represent argument bindings. Simply stated, the idea is just this: Corresponding to every argument of a predicate, have a distinct node in the network. Imagine every clock[4] cycle as divided into some number of slots, which we will refer to as *phases*. Associate distinct phases of the clock cycle with distinct objects participating in a reasoning episode[5]. Thus, in the example of reasoning given above, we would have chosen distinct phases, say, first, second, and third phases of clock cycles to correspond respectively to the objects $tweety$, $tree1$, and $tree2$. The binding of an object to an argument is represented by having the node corresponding to that argument become active in the phase associated with the object. More than one argument node can be active in a phase of a clock cycle denoting that the object associated with the phase is bound to all of those arguments.

Based on this idea of representing bindings, we developed a fairly powerful rule-based reasoning system[1,7]. This system, whose size is only linear in the size of the knowledge base, is capable of performing a large number of inferences involving rules and facts in parallel. The time taken to perform a single inference is proportional to the length of the derivation, and hence, optimal. However, as done in almost all connectionist reasoning systems developed so far, we had placed an important restriction on the kind of reasoning performed by the system. The

[2] Some ideas on how to render a degree of softness to the system thus arrived at, are mentioned later in the paper.

[3] Hence, the name *dynamic objects*, to objects represented that way.

[4] Actually, no global clock is necessary, as discussed in detail in [7]. However, for simplifying the discussion, here we are assuming that there is a global clock.

[5] Since the number of phases in a clock cycle is bounded, only a limited number of objects can participate in a particular reasoning episode. This restriction confirms with the psychological observation that human reasoning can focus only on a small number of objects at any time[6]. For detailed discussions of the neurological and psychological plausibilities of the synchronous activation approach to represent bindings, please refer to [7].

variable binding mechanism employed there was restricted to binding only *static objects* to arguments. For example, in the example of infering $move(tweety, tree1, tree2)$, and $reach(tweety, tree2)$ given above, the arguments of the relations are bound only to the static objects $tweety$, $tree1$, and $tree2$. An example of binding dynamic objects would be binding the dynamically created object $move(tweety, tree1, tree2)$ itself to an argument of some other other relation. The restriction that bound objects be static precluded us from having function symbols in rules and facts, since, to deal with function terms, the ability to bind dynamic objects to arguments is required. For example, consider the simple rule

walk-on(x,y) $\Rightarrow$*touch(feet-of(x),y)*

Now, suppose we have the fact *walk-on(john,street6)*. From this fact, we can infer the fact *touch(feet-of(john),street6)*. Notice that here, *feet-of(john)* is a dynamic object created during the reasoning process by binding the argument of the function symbol *feet-of* to the object $john$. In infering the fact *touch(feet-of(john),street6)*, the dynamic object *feet-of(john)* should itself be bound to the first argument of the relation '$touch$'.

In this paper, we describe how the synchronous activation approach can be extended to handle the above general kind of binding. That extension allows reasoning with function symbols and opens up interesting ways of enhancing the power of our earlier reasoning system, still retaining its nice features. Before we go into the details of these, we will digress a little to make a few comments on related research works.

To the best of our knowledge, no connectionist reasoning system has so far addressed the problem of reasoning with function symbols in any significant detail. Hence, our comments on related work is limited to the abilities of other proposed variable binding solutions to bind dynamic objects [6]. In [5], Lange and Dyer suggest the use of *signatures* to represent variable bindings. They permanently allocate a distinct signature to each static object and represent a binding by propagating the signature of the appropriate object to the argument to which it is bound. Now, to bind a dynamically created object to some argument, a signature will have to be recruited for the dynamic object on the fly during the reasoning process. Doing this appears very problematic. This is especially so, if signatures are learnt patterns, as suggested in [5]; that will mean that learning has to take place during an inference step. In [8], Smolensky addresses the variable binding problem in its abstract form and presents a tensor product based solution. He deals with the issue of nested bindings also. But, the way in which the proposed technique can be used in the context of reasoning remains to be seen. Barnden's 'Conposit'[2] is probably the only reasoning system other than the one presented in this paper, that is capable of handling nested bindings. However, the variable binding capabilities of Conposit come with a very significant sacrifice in knowledge level parallelism; in Conposit, only one rule can fire at a time. DCPS[9] is another reasoning system that is restrictive in its use of knowledge level parallelism. There does not appear to be any obvious way of extending the binding mechanism of DCPS to handle nested bindings.

2 Overview

Cognition requires both *forward reasoning* and *backward reasoning*. In forward reasoning, the system starts with a collection of facts and these facts trigger some rules leading to the inference of some other new facts. These newly derived facts in turn trigger some other rules and so on, the process leads to the derivation of a large set of facts that are deducible from the starting set of facts. A connectionist forward reasoning system dealing with function terms is discussed in some detail in section 3. That section describes the activity patterns chosen to represent function terms, the encoding of rules involving function symbols and how inferences involving function symbols is done in the network. Section 4 briefly sketches how a backward reasoning system incorporating function terms in rules and facts can be built. In that context, we indicate how long-term facts involving function terms (such as,

[6] Detailed comparisons of the synchronous activation approach with other proposed solutions, based on other relevant criteria, such as network size, reasoning speed, neurological plausibility etc., have been made elsewhere[7].

say, *hate(john,brother-of(tom))* are encoded in the network. Section 5 discusses some extensions to the work on reasoning with function symbols. One of the major extensions is that of combining the rule based reasoner with a parallelly operating equality reasoner.

3 A Forward Reasoning System that Deals with Function Terms

Before explaining how a reasoning system incorporating function terms is realized, let us discuss the dynamic representation of function terms.

A function term f of n arguments gets represented in the system using a relation R_f of $(n+1)$ arguments[7]. The first n arguments of R_f correspond to the n arguments of f respectively and the $(n+1)$th argument of R_f can be thought of as corresponding to the value of the function for those arguments. A dynamic object $f(c_1, c_2, ..., c_n)$[8] gets represented as follows: Recall from the discussion of section 1 that distinct objects participating in a reasoning process are associated with distinct clock phases. Suppose that the phases associated with the objects c_1,...,c_n are respectively p_1,...,p_n[9]. As said above, the first n arguments of R_f correspond to the arguments of f and hence are bound to c_1, c_2,...,c_n respectively. Using the synchronous activation approach, these bindings get represented by having the ith argument node of R_f become active in the p_ith phase of every clock cycle ($i = 1, ..., n$). Corresponding to the dynamic object $f(c_1, ..., c_n)$,[10] a free phase (i.e., a phase that is currently not assigned to any object) is assigned. The $(n+1)$th argument node of R_f is activated in this phase, thereby denoting that it is bound to the object $f(c_1, ..., c_n)$.

Summarizing, the following pattern of activity represents the function term $f(c_1, ..., c_n)$:

- The ith argument node of R_f becomes active in the p_ith phase of every clock cycle.
- A currently free phase (let it be p_*) is found and that phase is associated with the dynamic object $f(c_1, ..., c_n)$. The $(n+1)$th argument node of R_f will be active during the p_* phase of clock cycles.

Now, consider the problem of binding the object $f(c_1, ..., c_n)$ to an argument of some relation, say jth argument of the relation K. All that is to be done for this, is to activate the jth argument node of K in the phase p_* .

With that introduction, let us proceed to the encoding of rules involving function symbols. Consider the rule

$$P(x, y, z) \Rightarrow Q(f(y, x), z) \tag{1}$$

Associated with the relations P and Q, there are three and two argument nodes respectively (shown as diamond shaped nodes in Fig. 1)[11]. To represent the binary function f, a ternary relation R_f is used. The correspondences between the arguments of relations are denoted by the links between the appropriate argument nodes. For example, as per the above rule, the second argument of Q gets bound to the same object that binds the third argument of P. This is denoted by a link between the argument node a3 and the argument node a5 . Ignore the hexagonal box marked B for the time being.

Using the rule (1), starting from the fact $P(a, b, c)$, we can infer $Q(f(b, a), c)$. Let us see how this gets done with the rule encoding shown in Fig.1. Suppose that the objects a, b, and c are assigned the first, second, and third phases of clock cycles. Thus, the bindings in the starting fact are denoted by having the first, second, and third argument nodes of P active in the first, second, and third phases of clock cycles. The link from the argument node

[7] The reader is urged to remember the notation 'R_f'. In all the examples and figures, we use this notation.

[8] It is not required that c_is be distinct. In addition, c_is can be static or dynamic objects. Hence, nested function terms such as $f(g(h(a1, a2), a3))$ are allowed.

[9] If $c_i = c_j$, then, p_i will be equal to p_j.

[10] It may help to view this object as the value of the function f for the argument objects $c_1, ..., c_n$.

[11] We won't be explaining the role of the nodes drawn as pentagons in Fig.1. Due to space limitations, we have focussed on conveying the essential ideas and have omitted some details.

a3 to the node a5 causes a5 also to become active in the third phase of clock cycles - thereby, binding a5 to the object c, as desired.

The argument nodes a6 and a7 are connected respectively to the nodes a2 and a1. This causes a6 and a7 to become active in the second and first phases of clock cycles respectively. That is, a6 and a7 get bound respectively to b and a, thereby creating a part of the representation of $f(b,a)$. As discussed at the beginning of the section, now, a free phase in the clock cycle has to be found for the object $f(b,a)$. This is done by the hexagonal box labelled B, which we will refer to as *phase requester*. This box is not a single connectionist node; instead, it is a small piece of circuitry. There exists a global record maintainer that keeps track of the currently assigned phases. Box B communicates with this record maintainer to determine a free phase. We are suppressing the details of the phase requester and the record maintainer. For further discussions, it is assumed that the output of B is a pulse in the chosen free phase. The output of B goes to the third argument node of R_f. Thus, a8 becomes active in the phase chosen for the object $f(b,a)$. The link from a8 to a4, in turn, causes a4 to become active in the chosen phase. That is, the first argument of Q gets bound to the object $f(b,a)$. Now, we have the complete representation of the inferred fact $Q(f(b,a),c)$.

Above we discussed how a single rule is encoded and inference gets performed with that rule. Fig.2 shows how a collection of rules can be encoded. With this arrangement, an inferred fact in turn can spontaneously trigger other rules leading to the derivation of further facts. It is common to have one predicate occurring in the antecedent of many rules. When a fact involving a predicate is inferred, this triggers all those rules in which this predicate is the antecedent. Thus triggered rules derive further facts in parallel.

4 A Backward Reasoning System Dealing with Function Terms

Previous section described a *forward reasoner* incorporating function terms. Cognition also requires *backward reasoning*, where the task performed is essentially that of verifying whether a given fact is derivable from the rules and facts stored in the system. Two major tasks involved in performing backward reasoning are : (i) applying rules in the backward direction , (ii) checking whether a long-term fact is stored in the system.

The first of these tasks can essentially be achieved by having links in the reverse of the directions marked in Fig. 2 (A few other small changes will also be necessary.).

To see how the second task is performed, we would like to emphasize the two kinds of representations of facts: long-term facts are represented in the network as an interconnection of nodes while short-term facts get represented by a temporary pattern of activity. To check whether a particular long-term fact is present in the system, the pattern of activity corresponding to the short-term representation of that fact is induced in the network. The interconnection encoding a long-term fact is such that the activity of a particular node in the interconnection goes high when and only when the activity corresponding to the short-term representation of that fact is present in the network[12]. Such an interconnection corresponding to the long-term encoding of the fact $P(a,f(b))$ is shown in Fig.3. In figures, we use links with dark circles at their ends to denote inhibitory connections and links with solid arrows at their ends to denote enabling connections. Nodes marked with the names of the constants a and b are referred to as *constant nodes*; they will be active in the respective clock phases assigned to these objects in a reasoning episode. Node marked N is a coincidence detector; it sends a high output if and only if all of its inputs are active and synchronous. The node marked K sends a continuous high output. Node M, drawn as a horizontal pentagon is a *temporal-AND* node, which becomes active if and only if it receives activation throughout a clock cycle. Recalling the short-term representation of function terms discussed earlier, one can easily verify that the condition for M to become active gets satisfied when and only when the pattern of activity corresponding to the short-term representation of the fact $P(a,f(b))$ is present in the network.

[12]Thus, the activity of this node can be used in answering the question whether a particular long-term fact exists or not.

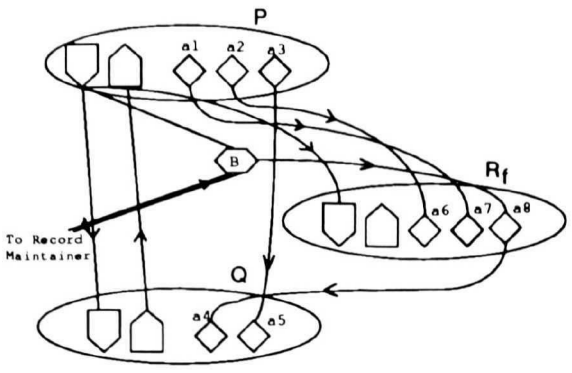


Figure 1: Encoding of the rule $P(x, y, z) \Rightarrow Q((f(y, x), z))$.

Using the above discussed technique for encoding long-term facts involving function terms, a backward reasoning system can be realized along the lines described in [1,7].

5 Extensions of the Work

5.1 Combining with an equality reasoner

In previous sections we discussed how a rule-based reasoning system incorporating function terms can be realized. Such a rule-based reasoner can be elegantly interfaced with a parallelly operating equality reasoner to obtain a more powerful reasoner. The operation of these two components is as follows: The main reasoner derives new facts and the equality reasoner substitutes equivalent terms for the terms occurring in those facts to derive some other facts. Those new facts derived by the equality reasoner can in turn trigger rules in the main reasoner. Here, we will only sketch some of the ideas involved in the realization of such a hybrid reasoner and will report the details elsewhere.

The equality reasoner operates with two kinds of equalities. Substitutions involving these two types are accomplished in different ways. The first type of equality states that an entity described using a function symbol is same as a known domain individual. An example of this kind of equality is *maternal-uncle-of(tom)=dave*. Reasoning with this kind of equality is accomplished by extending the mechanism for answering *wh-queries* reported in [7].

The second kind of equality states equalities between two entities both of them being described using function symbols. An example of this kind of equality is $\forall x$ *brother-of(mother-of(x))=maternal-uncle-of(x)*. A piece of circuitry to apply this equality to substitute *maternal-uncle-of(x)* for *brother-of(mother-of(x))* is shown in Fig. 4. This is accomplished by viewing this operation as the application of the rule *brother-of(p,q)*$\wedge$*mother-of(q,r)* $\Rightarrow$*maternal-uncle-of(p,r)*. Network realization of this is analogous to the encoding of conjunctive antecedent rules described in [1,7] and is indicated in Fig.4. There, the rectangular shaped node marked B is a coincidence detector whose output will be high if and only if its two inputs are active and synchronous. This node ensures that the same object binds the second argument of *brother-of* and the first argument of *mother-of*, as required by the above rule.

In the derivation of a particular fact both of the above two types of equalities may be made use of. Thus, if the main reasoner derives *live-in(brother-of(mother-of(tom)),boston)*, then the network may first derive *live-in(maternal-uncle-of(tom),boston)* using the second kind of equality and then derive *live-in(dave,boston)* using the first kind of equality.

5.2 Evidential reasoning

As mentioned in section 1, our primary focus so far has been on exploring how reasoning with logical rules and facts can be efficiently realized in a connectionist network. In achieving this, we assumed all the link weights to be unity. A certain degree of softness can be rendered to this reasoning system by making the rules to be probabilistic and associating a measure of certainity with the facts. These modifications can be easily realized in the network architecture developed, by having non-uniformly weighted links.

5.3 Multiple Dynamic Facts

In the description of the system presented in this paper, we have assumed that only one dynamic fact involving a relation is present at any time. Though this is a common assumption made in connectionist reasoning systems[4,5], there are situations in which this restriction is unacceptable. In [7], we discuss some techniques of relaxing this restriction in the case of our system.

6 Conclusion

We described a connectionist rule-based reasoning system incorporating function terms. The system is extremely efficient both in network size and the time taken to perform inferences. The extended variable binding capability introduced in the system opens up many interesting avenues for realizing *broad, yet, efficient, reasoning* - the characteristic feature of human cognition. They are currently being investigated.

References

[1] Ajjanagadde, V.G., and Shastri, L., Efficient inference with multi-place predicates and variables in a connectionist system, *Proceedings of the Cognitive Science Conference*, August 89, pp. 396-403.

[2] Barnden, J., Neural-net implementation of complex symbol-processing in a mental model approach to syllogistic reasoning, *Proceedings of IJCAI-89*, pp. 568-573.

[3] Feldman, J.A., Dynamic connections in neural networks, *Bio-Cybernetics*, 46:27-39, 1982.

[4] Hinton, G., Implementing semantic networks in parallel hardware, In G.Hinton and J.Anderson (Eds.),*Parallel Models of Associative Memory*, Erlbaum, 1981.

[5] Lange, T., and Dyer, M., High-Level inferencing in a Connectionist Neural Network. Technical Report, UCLA-AI-89-12, October. Computer Science Department, UCLA, 1989.

[6] Miller, G.A., The magical number seven, plus or minus two: Some limits on our capacity for processing information, *The Psychological Review*, 63(2), March 1956, pp. 81-97.

[7] Shastri, L. and Ajjanagadde, V., From simple associations to systematic reasoning: A connectionist representation of rules, variables, and dynamic bindings, Tech. Report, Dept. of Computer Science, Univ. of Pennsylvania, January 90.

[8] Smolensky, P., On variable binding and the representation of symbolic structures in connectionist systems, Technical Report CU-CS-355-87, Department of Computer Science, University of Colorado at Boulder, 1987.

[9] Touretzky, D. and Hinton, G., A Distributed Connectionist Production System. *Cognitive Science*, 12(3), pp. 423-466.

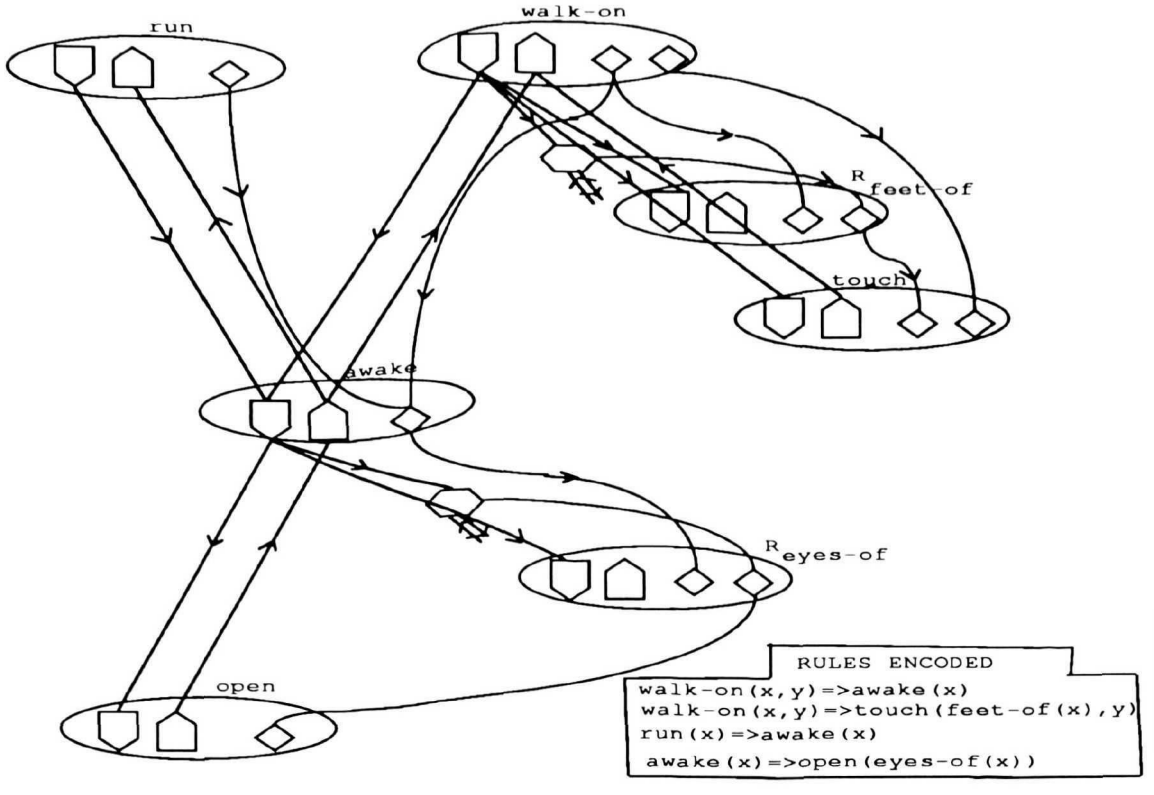


Figure 2: An example network.

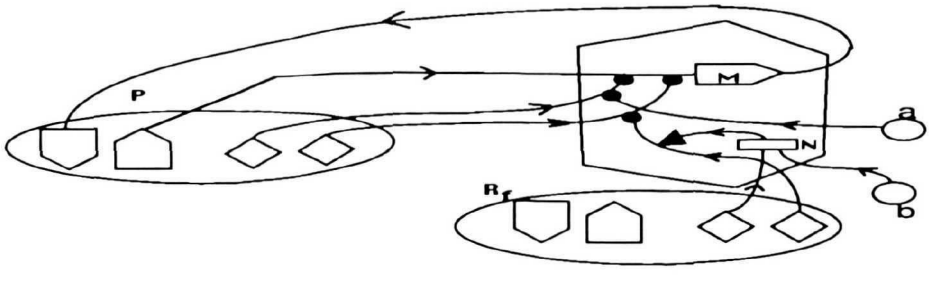


Figure 3: Encoding of the long term fact $P(a, f(b))$.

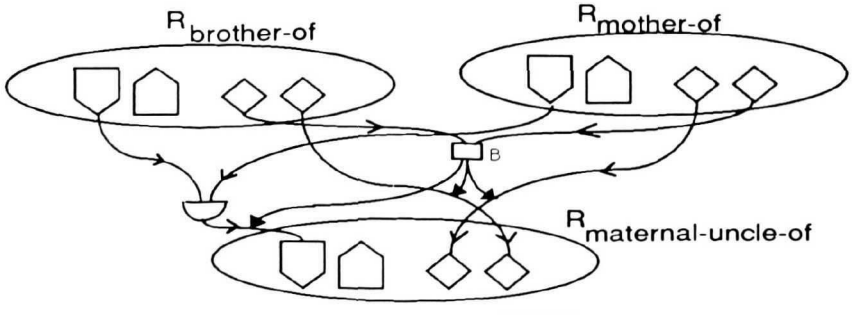


Figure 4: Interconnection for substituting $maternal\text{-}uncle\text{-}of(x)$ for *brother-of(mother-of(x))*.

NOT ALL POTENTIAL CHEATERS ARE EQUAL: PRAGMATIC STRATEGIES IN DEDUCTIVE REASONING

VITTORIO GIROTTO (*CNR, Rome, Italy*)
PAOLO LEGRENZI (*University of Trieste, Italy*)

ABSTRACT

This work briefly discusses one of the central problems in the current psychology of reasoning: that of explaining the effects of content. Two competing theories recently proposed to explain such effects (pragmatic reasoning schemas and social contract theories) are illustrated with reference to an experiment on reasoning in children employing a selection problem, which requires a search for the potential counterexamples of a conditional rule. On the one hand, the theory of pragmatic schemas (i.e. clusters of rules related to pragmatically relevant actions and goals) predicts that correct selection performance derives from the activation of specific contractual schemas, such as obligation and permission, the production rules of which correspond to the logic of implication. On the other hand, according to the social contract theory, people are able to detect potential counterexamples only when they correspond to the potential cheaters of rules having the form 'If benefit A is received, then cost B must be paid'. The results of the experiment show that performance on tasks of this kind is not determined simply by the possibility of representing the rule in question in cost-benefit terms; to predict performance one necessary factor is knowledge of the nature of the possible cheating behaviour that one is requested to check.

ACKNOWLEDGEMENTS. Preparation of this paper was supported by CNR grants. Correspondence should be sent either to V. Girotto, Istituto di Psicologia, CNR, Viale K. Marx 15, 00137, Roma, Italy (e.mail: vittorio@irmkant.bitnet) or to P. Legrenzi, Dipartimento di Psicologia, Via dell'Università 7, 34127, Trieste, Italy).

The ability to search for counterexamples has a central role in reasoning, since the ability to search for them can be regarded as the basis of the discovery and evaluation of hypotheses, concept attainment, and deductive inferences. However, a great deal of empirical evidence exists that adults perform poorly in reasoning problems requiring a search for potential counterexamples, for example in general statement evaluation problems. Studies that utilized the well-known Wason four-card selection task (Wason, 1966, 1968), showed that the majority of adult subjects did not search for counterexamples to a rule such as "If a card has a vowel on one side, then it has an even number on the other side". Most adults typically fail to select a card with an odd number, one of the potentially falsifying cards. In general terms, the rule used in such problems is a universal statement, typically a conditional statement, *if p then q,* and the relevant cases are *p and not-q*, in the above indicated example "a card with a vowel" and "a card with an odd number".

The long tradition of research with tasks of this type has shown that in some cases people are able to search for counterexamples, particularly when problems are phrased in "concrete" terms (Johnson-Laird, Legrenzi and Sonino-Legrenzi, 1972; for reviews see Griggs, 1983; Wason, 1983).

Among the different proposals for explaining the ensemble of findings produced with selection problems, the most convincing seems to be the pragmatic reasoning schemas interpretation (Cheng and Holyoak, 1985). Pragmatic schemas, such as permissions, obligations and causations, are clusters of rules which concern pragmatically relevant actions and goals. Under certain circumstances, some of these schemas lead to the correct solution of problems demanding a search for counterexamples. In particular, the activation of a permission or obligation schema can help people to solve a selection problem. Although their production rules go beyond those of the logic of material implication (for example, by including modal verbs such as "must" and "can"), their productions lead to card selections which *correspond* to those prescribed by a logical analyis of the task. For example, a permission rule *'If you want to do action A, then you have to satisfy precondition B '*, implies the contrapositive rule *'If you do not satisfy precondition B, then you are not allowed to do action A '*. This equivalence makes clear that the potential violators of the permission rule are people who have done action A without satisfying precondition B.

Cheng and Holyoak (1985) obtained empirical findings that corroborate the pragmatic schemas hypothesis. They have shown

that adult subjects are able to search for potential violators of unfamiliar but rationalized permission rules, and that succesful performance is also elicited with an abstract description of a permission situation. Moreover, facilitation of selection performance in conditions concerning permission and obligation rules has been obtained with preadolescent children (Girotto, Light and Colbourn, 1988).

A different interpretation of the content effect in reasoning performance has been recently proposed by Cosmides (1989). According to her "social contract theory", people process information regarding social exchanges using specific, naturally evolved algorithms. In particular, social contract algorithms express an exchange in which an individual is obliged to pay a cost in order to be entitled to receive a benefit. They contain an inferential procedure (*'look for cheaters* ') that enables people to detect potential cheaters (i.e. individuals who have not paid the required cost, and individuals who have accepted the benefit). In a series of experiments, Cosmides (1989) showed that wording a selection problem in terms of a social contract (*If one accepts benefit A, then one has to pay cost B*) can produce formally correct performance when the cases indicating possible cheating ('benefit accepted' and 'cost not paid') correspond to the formally relevant cases, i.e. the potential counterexamples.

There has been a lively and still open debate between the two described positions. According to Cosmides' theory, only social contracts, which are a subset of all permission rules, produce "robust and replicable content effects" on selection tasks.

Cheng and Holyoak (1989) have criticized this position, in which only the cost/benefit representation is considered to be psychologically real. Moreover, a number of empirical studies have shown that correct reasoning performance, both for adult and child subjects, can be obtained in conditions which, following social contract theory predictions, should not activate the described 'look for cheater' procedure This has been the case, in particular, of certain prudential (Cheng and Holyoak, 1989; Girotto, Gilly, Blaye and Light, 1989; Manktelow and Over, 1990), obligation (Girotto, Blaye and Farioli, 1989) and permission (Light, Girotto and Legrenzi, in press) rules which did not directly map the cost/benefit structure of standard social contracts.

In the present paper, we will briefly present the results of a research about children's reasoning on conditional promises and permissions (reported in detail in Light *et al.*, in press). We will discuss their theoretical implications in relation to the indicated debate.

Consider a conditional contractual promise, like the following, made by a teacher to her pupils:

"*If you get at least 10 points, then you can have a sweet* "

And suppose that there are four pupils:

Mary (who had 10 points), Ben (4 points), Sue (who has received a sweet) and Rob (who has not received a sweet).

Clearly, in this scenario it is unlikely that the promisor (the teacher) will violate her own promise (by not giving the reward to the deserving pupils).

A more likely outcome is that some promisees will try to cheat by taking a reward which they do not deserve.

A teacher-promisor who decides to check whether her promise has been respected should thus make sure that Ben (the pupil with 4 points) and Sue (the pupil with the sweet) have not cheated, that is, that they have not respectively taken a sweet and obtained less than 10 points.

If this checking condition is considered as a version of the selection task, it is clear that the pragmatically correct choice will be that of examining the two pupils just mentioned (Ben and Sue), whose formal values are respectively *not-p* and *q* (which are different from those indicating the potential counterexamples of a conditional rule *'if p then q* ', namely the values *p* and *not-q*).

Now, if we consider a condition in which the teacher-promisor has delegated a specific pupil to administer the promise, this agent may commit two types of infraction. If the agent acts selfishly, he will tend to withhold the cake from the pupil who deserves it. For this reason, a teacher-promisor wishing to check the agent's behaviour should ensure that Mary (the pupil with 10 points) and Rob (the pupil with no sweet) have not been unfairly deprived of the reward. If instead the agent has behaved nepotistically, then the teacher-promisor will have to check not only the two pupils just indicated but also the others (Ben and Sue), who might have received the reward, although underserving, because they were friends of the agent. In the first case (selfish agent) the formal values of the cases to be checked are *p* and *not-q*; in the second case (nepotistic agent) all four values must be checked: *p, not-p, q, not-q*.

As can be seen, for a conditional contractual promise there can be various possible combinations of cases that indicates a violation. The formal values of these cases do not always correspond to the combination *p* and *not-q*.

Three versions of a selection task concerning these three possible situations of promise violation were presented by Light *et al.* (in press) to some English children aged 11-12 years (a fourth condition, which serves as a control, concerned a permission rule "*If you want a sweet, then you must get at least 10 points* ", which could be violated by Sue, formally *p*, and Ben, formally *not-q*).

The results of this research have shown that preadolescent children do master the complex pragmatic factors underlying the control of conditional contractual promises and permission. Their patterns of responses correspond, in most cases, to the selection of the different combinations of the potential violations above indicated: In the condition where the pupils-promisee could violate the promise by themselves ('direct promise' condition), the most frequent choice (50%) was the selection of the cards *not-p* and *q*. In the two conditions where the teacher-promisor had to check agent's behaviour, children' s selection turned out to be different. In the 'selfish agent' condition, the *p* (10 points) and *not-q* (no sweet) cards were indeed selected most regularly (90% and 83% respectively). However, only 22% of the subjects selected just these two cards. In addition, 'sweet' card (*q*) continued to be selected by many children. In the 'nepotistic agent' condition, the prevalent choices was the combination of *all cards* (39%). Finally, in the permission condition, the correct pattern *p* and *not-q* was most frequently selected (78%).

It is possible to compare, although indirectly, these data with those reported by Cosmides (1989). Some of the social rules used in her experiments are in fact conditional contractual promises. For example, in two experiments, her subjects had to check the behaviour of the promisor (an African hunter, called Bo) of the following deal "*If you give me your ostrich eggshell, then I'll give you duiker meat* ". This condition is similar to Light et al. 's '*selfish agent* ' condition: In both cases, in order to detect the possible cheaters one has to check whether the deserving promisee (pupil with 10 points or man who gave Bo eggshells) had received the earned reward (sweet or meat), excluding, at the same time, that the persons who actually ran the deal (the selfish agent or the promisor himself, Bo) had illegally kept it. In other words, in both cases, the relevant cards correspond to *p* and *not-q*. Now, despite this similarity in the structure of the two problems, the elicited performance turned out to be different. While Cosmides' scenario elicited about 70% of *p* and *not-q* selections, in Light et al. 's selfish agent condition, this pattern of response was produced only by 22% of the subjects.

If we compare the ways in which the runners of the deal are presented in the two scenarios, several differences seem to appear.

In the story used by Cosmides, the subjects had to check a promisor (Bo) who was, at the same time, 1) *owner* of the goods (duiker meat) that he should have given to the others in exchange for the fulfillment of the contractual requirements; 2) *motivated to keep* for himself the maximum amount of these goods (Bo was actually presented as an "unscrupulous man...(who) had very little duiker meat and a large family to feed"); 3) personally *motivated to obtain* the fulfillment of the contract (Bo was presented as someone who was "always accidentally breaking his ostrich eggshells and would like to 'stockpile' some").

In Light et al. 's 'selfish agent' condition, the agent of the promise, 1) was *not* the *owner* of the goods (sweets) that he should allocate to the deserving pupils; 2) even if he was allowed to keep for himself the non allocated goods, it was *not specified* whether he was really *motivated* to do so (i.e. whether he was a glutton); 3) he was *not* personally *interested* in obtaining the fulfillment of the contractual requirements (i.e. the successful school performance of the classmates); 4) his *relationships* with the promisees were *not* specified (i.e. he could be a good friend or an enemy of the other pupils).

Thus, while in the Cosmides' stories it was clearly specified that the promisor was motivated to a selfish cheating behavior, in Light et al. 's condition the agent of the promise could plausibly behave both nepotistically and selfishly. This possibility to attribute different goals to the agent can explain why Light et al. 's children did not limit themselves to the selection of the 'selfish' cards *p* and *not-q*.

This comparison shows the importance of the *information about the nature of the cheating behaviour* that one is requested to check. Both children (selfish agent condition) and adults (cf. Politzer and Nguyen-Xuan, 1988) seem to have difficulties in performing consistently this check when this information is not sufficient. However, it should be noted that in conditions where sufficient information about the goals of the possible cheaters is given, children can produce consistent selection performance. This was the case of the permission and direct promise conditions in Light et al. 's study, where the nature of the potential cheating behaviour could only be 'selfish' (as the pupils acted alone). In the former case, children consistently (78%) checked the two pupils who could have violated the permission rule (the pupil with the sweet and the pupil with 4 points, i.e. *p* and *not-q*). Children (50%) still consistently selected these two cases in the latter condition, even if their logical

values (i.e. *not-p* and *q*) where different from those of the cards selected in the permission condition. This specific response pattern is similar to that obtained by Cosmides (1989) in versions of the task in which a contractual promise (social contract) was modified to a sort of obligation (switched social contract). For examples, the original Bo's promise ("*If you give me your ostrich eggshell, then I'll give you duiker meat* ") was modified to "*If I give you duiker meat, then you must give me your ostrich eggshell* ". In this case, people had still to check Bo's behaviour, and they still selected the cards corresponding to 'benefit for Bo' and 'cost unpaid by Bo', which have formal values (*q* and *not-p* , respectively) different from those of the original condition (*p* and *not-q*).

In conclusion, the results presented by Light *et al* (in press) show: a) that, regardless of the similarity of rules and scenario, reasoning performance can dramatically change as a function of the *actor* who could have infringed a conditional promise (and his/her goals); b) despite the possibility of recognizing a situation as one of social exchange (*sensu* Cosmides), subjects do not consistently look for potential cheating behaviour, when complete information about it is not provided. Therefore, while previous research has demonstrated that succesful reasoning performance can be obtained even in conditions which cannot be represented in the cost/benefit terms of a social contract, Light *et al* 's study demonstrates that conditions which can be represented in these terms do not necessarily elicit the pattern of responses predicted by the social contract theory.

REFERENCES

Cheng, P.W., & Holyoak, K.J. (1985). Pragmatic reasoning schemas. *Cognitive Psychology, 17*, 391-416.

Cheng, P.W., & Holyoak, K.J. (1989). On the natural selection of reasoning theories. *Cognition, 33* , 285-313.

Cosmides, L. (1989). The logic of social exchange: has natural selection shaped how humans reason? Studies with the Wason selection task. *Cognition, 31* , 187-296.

Girotto, V., Blaye, A., & Farioli, F. (1989). A reason to reason. Pragmatic basis of children's search for counterexamples. *European Bulletin of Cognitive Psychology, 9*, 297-321.

Girotto, V., Gilly, M., Blaye, A., & Light, P.H. (1989). Children's performance in the selection task: Plausibility and experience. *British Journal of Psychology, 80* , 79-95.

Girotto, V., Light, P. H., & Colbourn, C. (1988). Pragmatic schemas and conditional reasoning in children. *Quarterly Journal of Experimental Psychology, 40A* , 469-482.

Griggs, R.A. (1983). The role of problem content in the selection task and THOG problem. In J. St B.T. Evans (Ed.) *Thinking and reasoning: Psychological approaches.* London: Routledge and Kegan Paul.

Johnson-Laird, P.N., Legrenzi, P., & Sonino-Legrenzi, M. (1972). Reasoning and a sense of reality, *British Journal of Psychology, 63*, 395-400.

Light, P.H., Girotto, V., & Legrenzi, P. (in press). Childen's reasoning on conditional promises and permissions, *Cognitive Development.*

Politzer, G. & Nguyen-Xuan, A. (1988). Pragmatic reasoning schemas: Promises and the four-card selection task. Unpublished manuscript, Centre National de la Recherche Scientifique, Paris.

Wason, P.C. (1966). Reasoning. In B. Foss (Ed.), *New horizons in psychology,* Harmondsworth: Penguin Books.

Wason, P.C. (1968). Reasoning about a rule. *Quarterly Journal of Experimental Psychology, 20*, 273-281.

Wason, P.C. (1983). Realism and rationality in the selection task. In J. St B.T. Evans (Ed.), *Thinking and reasoning: Psychological approaches.* London: Routledge and Kegan Paul.

Explanations in Cooperative Problem Solving Systems

Thomas Mastaglio and Brent Reeves

Department of Computer Science and
Institute of Cognitive Science
University of Colorado, Campus Box 430
Boulder, CO 80309
email: brentr@boulder.colorado.edu

Abstract

It is our goal to build cooperative problem solving systems, knowledge-based systems that leverage the asymmetry between the user's and the system's strengths and thus allow the dyad of user and computer system to achieve what neither alone could achieve. Our experience has shown that in these cooperative systems, the need for explanations is even more evident than in traditional expert systems. This is due to the fact that these new systems are more open-ended and flexible and therefore allow for more possibilities in which a user can reach an impasse, a point at which it is not clear how to proceed.

Observation of human-human problem solving shows that people are sensitive to the domain under discussion and the other's knowledge of that domain. People tend to construct explanations that are minimal in the number of concepts or chunks. These explanations are not comprehensive, and the communication partner is able to follow up on aspects which are still unclear.

1 Introduction and Theoretical Basis

Our research focuses on how to build cooperative knowledge-based systems that take advantage of the different strengths of users and computer systems. Computers can be a source of expert domain knowledge, knowledge they can use to make suggestions to users. The computer system's role in this dyad must include the ability to explain those suggestions. For various reasons, current explanation systems often fail to satisfy users. Too frequently they are based on an implicit assumption that explaining is a one-shot affair and that the system will be able to produce or retrieve a complete and satisfying explanation provided it is endowed with *artificial intelligence*.

Our approach takes advantage of existing information and knowledge-based systems technology already available to provide the user access to explanations at different levels of detail and complexity. Development efforts focused on the concepts requiring explanation rather than on selecting a complete prestored explanation. A conceptual model of the domain and a user model provide that set of concepts. Explanation giving requires establishing a suitable conceptual context for a dialogue between the user and the system. The system must know what concepts are required to understand an entity in the domain and furthermore which subset of those concepts are unfamiliar to a given user. Forming such an idiosyncratic explanation is a construction rather than a selection process.

We are investigating how to design systems that serve users as they are actively engaged in their own work — cooperative problem solving systems. These systems provide a task-based environment which users employ to accomplish some goal. We want them to be more than a communication medium with which to describe a problem. In the context of these task-based environments we analyzed the

reasons people seek explanations. The common triggering condition is that they experience an impasse that stops or limits their work. We call these "task-oriented impasses" and have cataloged them to better understand how explanation giving can help to overcome the impasses. The four categories of task-oriented impasses are: action impasses, communication impasses, motivation impasses, and curiosity impasses.

1. *Action impasses* occur when users do not know what to do next. Some action impasse questions are: What should I do next? Is *action* the right thing to do next? How do I do *action*? What did the system just do? What are the results of doing *action*? Can I do *action* now?

2. A *communication impasse* is a failure to understand a given object in the environment. Representative questions are: What is *object*? Why is *object1* shown instead of what I expected, namely *object2*?

3. *Motivation impasses* fall into the realm of behavioral psychology; their basis is an anthropomorphic view of the computer system: Representative questions are: Why did the system do *action*? Why did the system just communicate with me? Why did the system just say *X*? Why should I do *action*? Why is $action_1$ better than $action_2$?

4. *Curiosity impasses* are a bit different. The other categories consist of questions that arise when users encounter a problem. Curiosity impasses are not necessarily impasses, in a strict sense, but rather a diversion from continuing with the task. They are circumstances in which users seek to gather information that is interesting or might be helpful, but *the lack of which* will not preclude them from further work. For consistency we will refer to them also as "impasses". Questions that illustrate curiosity impasses are: Is *object* a concept *X*? How do $object_1$ and $object_2$ differ?

To overcome these impasses, explanations in cooperative problem solving systems serve four functions; functions that are similar to findings of empirical work on explanation giving in expert systems [16]. The functions of explanations in cooperative problem solving systems are:

1. To allow users to examine the system's recommendations,

2. To relate recommendations to domain concepts — understanding "what is suggested",

3. To provide a rationale for recommendations — understanding "why this would be better",

4. To educate a user about underlying domain concepts.

These four functions are not mutually exclusive. A single explanation episode by a cooperative problem solving system will often accomplish several of these functions.

Even in cases where explanation systems have been designed with the above functions in mind, the result has typically been a system that explains too much, and therefore not enough. A familiar example to many computer system users is the *man page*, which contains everything a user might want to know, from the system's perspective, and for that very reason is often too much to be of use.

In observing problem-solving interactions between salesmen and customers in a large hardware store, we noticed that explanation never took the *man page* approach. When explanations were required, the approach was one of minimizing the explanation, then following up on unclear concepts when necessary [13]. This is interesting if you consider the fact that the particular store carries over 350,000 different items in over 33,000 square feet of retail space. If salesmen took the approach found in many computer systems, the explanations they gave would be extraordinarily long, in order to be complete, and complex, in order to take into account the relationships to other items in the store.

As an example of computer systems with extensive documentation, consider the Symbolics Lisp machine, which comes with over 4000 pages of documentation, most of which is available online in hypertext format. Issuing a query (i.e. a request for an explanation) can result in pages and pages of text and examples; frequently too much to digest and make sense of. As computer systems increase in power, they also increase in complexity. Simply having more powerful computers will not solve the problem of explanation, it will only exacerbate it.

In order to model the "minimalist explanation" approach found in human-human interactions in a knowledge-based system, we designed an extension to LISP-CRITIC, a knowledge-based source transformation system. Supporting idiosyncratic explanation giving requires a user model that contains a detailed representation of users' knowledge. The explanation approach needs something more than classifying users by level of expertise. Our framework uses a fine-grained user model and a minimalist approach [6]. More theoretical basis for the minimalist explanation approach is found in related work on discourse comprehension:

1. Short-term memory is a fundamental limiting factor in reading and understanding text [2, 1]. The best explanations are those that contain no more information than absolutely necessary, since extra words increase the chances that essential facts will be lost from memory before the entire explanation is processed.
2. It is important to relate written text to readers' existing knowledge [11, 5].

Similar guidelines are also offered as principles of rhetoric. Flesch developed formulae to evaluate the readability of text [9] that emphasize brevity. Computer explanation systems should comply with similar standards, therefore short sentences and known vocabulary as important criteria. Other support is found in Strunk and White's familiar manual; it contains similar advice. Writers of explanatory text are told "Don't explain too much" [15].

2 An Explanation Approach

Supporting explanation requires the system to have enough knowledge to describe what is going on and why. Operational knowledge is often captured in rule form (If *condition* then *action*). For users to understand a rule means that they must know the concepts underlying a rule. A conceptual domain model provides the entire set of prerequisite knowledge and a user model filters that set for an individual.

The user model helps select which subexplanations to actually present. One view of explanation is that users are engaged in an understanding process that is based on generating explanations to themselves [14]. It follows that the explanation component has to provide users with sufficient information so they can develop their own self-explanation, and yet not with so much information that it interferes with a person's ability to construct meaningful explanations.

Determining what to explain to a user requires decomposition; the system must execute a GPS-like process [3]. Prerequisite knowledge is the knowledge a user must know in order to understand a given concept. This is a recursive process, understanding those domain concepts that are prerequisites for the given concept requires, in turn, understanding their prerequisites and so on. To support such an approach a deep structure for the domain is queried to obtain the set of prerequisites. Still, a satisfactory explanation approach will do more, namely identify the concepts in that set that do not need explaining because the user already knows them. Then the system reasons about the best way to explain the remaining concepts. Based on this general schema, we developed a framework consisting of several

levels of explanations.

It is also important to incorporate a fallback capability, allowing the user some recourse when the initially provided explanation fails. No computer-generated explanation system can be expected to satisfy users completely — people are not able to achieve this and we do not expect any more of computer systems. The situation where a user does not understand an initial explanation or wants more detailed information must be accommodated. One technique is the reactive approach [12]; our approach is simpler in that it makes use of existing hypertext capabilities.

3 LISP-CRITIC Explanation System

We have developed an explanation component for a LISP critiquing system in accordance with this framework. LISP-CRITIC is that system [7, 4]. It is a knowledge-based interactive system that operates in a programmer's editing environment, providing suggestions on how to improve user LISP programs. When called on to examine a piece of code, the system informs the programmer of each suggestion it has for improving the program. As each suggestion is provided, the programmer can accept the suggestion, reject it outright or ask for an explanation. Users seek explanation to help them understand the suggestion well enough to make the accept/reject decision. When a suggestion is accepted the system actually rewrites that piece of code in the user's editing buffer.

We developed an approach that provides information in four layers of increasing detail. The first two layers are not necessarily explanations in a specific sense of the word but we found that in many situations they provide enough information to satisfy system users.

1. A fundamental piece of information is the name of the rule that identifies a suggested transformation. The rule name is an abstract reference to a chunk of domain knowledge, that chunk is an instance of the domain object class LISP-CRITIC rule; it may or may not have meaning to users. Sometimes when it does have meaning, users are satisfied just knowing which rule fired and no further explanations is required. They have sufficient information on which to base an accept/reject decision.

2. The transformed code, as recommended by the system, can be compared to the user's code. The system shows the user's code and its recommended version. Often this is sufficient for users to understand the suggestion and they can make a decision on whether to accept or reject the advice.

3. The minimal explanation layer is the point where our theoretical investigations comes into play. The user is provided with a textual description of the system's advice based on the domain concepts behind that transformation.

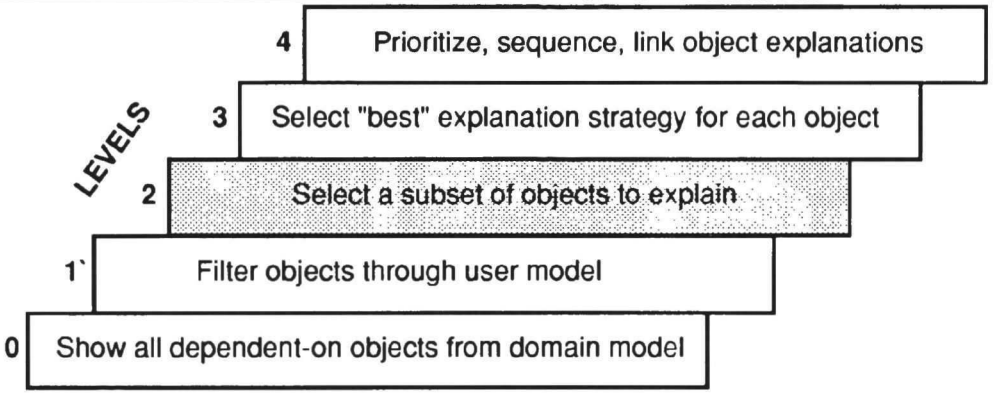
4. The underlying computational environment already includes a hypertext-based information space, the Symbolics Document Examiner. LISP-CRITIC facilitates access to this information for users who are not satisfied with the minimalist explanation of the advice. Users navigate through the hypertext space themselves; the system first locates them in an appropriate context.

The approach used in the current implementation evolved from attempts to provide explanations for an earlier version of LISP-CRITIC [10]. The approach taken involved rule-tracing and prestoring textual descriptions for each transformation (layer 1 and 2 above). A suite of alternative canned explanations for each rule were provided; each one designed to meet the needs of a user with a particular level of expertise. To provide the correct explanation, the system classified a user as a novice, intermediate or expert programmer. This approach was of limited success. A major finding from that work was the need

for a finer grained approach to modelling individual users than classification: An approach that also supports dynamic update as users' expertise changes.

The theoretical framework discussed above led us to emphasize concise and readable explanations. Metrics with which to evaluate, or guidelines to help construct such explanations do not exist. We decided that it would be best to provide terse explanations tailored individuals, while recognizing that at times users require additional information. To provide that information, we provided the hypertext "hooks" into an existing on-line documentation system.

The explanation levels shown in Figure 1 capture the necessary and sufficient conditions for an adequate explanation, each level incrementally improving on the work done at a lower level by using additional knowledge about the user and the domain. A level 0 explanation does not require knowledge about individual users. It uses the conceptual domain model to meet the necessary condition that it knows everything required to understand the entity needing to be explained. The set of prerequisite concepts required in order to understand the object the user wants explained is provided by a deep domain model. Level 1 brings the user model into the process; the prerequisite concepts are "filtered" through the user model to determine the subset of them appropriate for explaining to a given individual. What remains is often larger than what would be reasonably explained in a single dialogue episode. Therefore the explanation component uses knowledge about what makes a good explanation at level 2 to order and prioritize the concepts that are to be explained.



Five levels of explanation have been identified. Level 0 insures that all prerequisite knowledge for a given domain object is available to the explanation component. Level 1 builds on top of level 0 and so forth. The current implementation provides level 2 explanations. The domain and user model can also support levels 3 and 4.

Figure 1: Explanation Levels

Level 3 makes use of additional domain knowledge or perhaps other information in the user model to determine a "best" strategy for explaining each concept. For example, a system could make use of the links between objects in the domain model and the user model to determine candidate concepts or functions for use in a differential description [8]. One object can be described differentially in terms of another object that the user already knows. Level 4 performs "syntactic sugaring". Here the individual explanations from level 3 are ordered and linked in an appropriate manner. This is a nontrivial process

and requires the system to have knowledge of discourse as well as natural language generation capabilities that exceed the present state of the art.

4 Role of the user model in explanations

Cooperative problem solving systems must tailor explanations in order to adequately perform the four functions of explanation. The basis for this tailoring is the user model. The user model is an essential component. A simple approach is to classify users by their expertise (e.g., novice, intermediate, expert). The use of stereotyping and classification schemes can accommodate those systems that provide one-shot advise but cooperative problem solving systems need a finer grained representations of the user.

The user model contains a representation of the user's domain knowledge adequate to support any of the five "levels" of explanation shown in Figure 1. The implication is that it must be based, at least partially, on the conceptual model of the domain in order for it to serve as the filter for level 2 explanations. The model needs to capture the user's goals in order to support level 3 explanations.

Supporting level 4 explanations are more difficult. Explanations at this level are at the frontier of research on natural language generation and human-computer dialogue technology. We will not know everything required of a user model to support explanations at this level until that research matures. At this point we conjecture various user model capabilities needed to support this level, such as knowing the education and reading comprehension level of a user.

5 Summary and Conclusions

The problems with current explanation systems are widely recognized, but most efforts to improve them attempt either to emulate human-to-human communication in inappropriate ways or to provide a complete explanation in "one shot". Theoretical results in discourse comprehension and principles of rhetoric are a suitable starting point but it must be kept in mind that all human-to-human communication techniques are not appropriate for computer-based explaining.

In studying human-human problem solving, we note that one aspect that is less important than expected is the ability to produce and understand natural language. People rarely talk in grammatically correct or even complete sentences, yet they manage to communicate and solve problems. We call this 'natural communication,' rather than natural language. We have attempted to map natural communication over to human-computer systems by supporting minimalist, layered explanations with the capability for further follow-up via hypertext.

The approach we used provides several layers of explanation for advice from a knowledge-based system. The first two layers are not explanations in the strictest sense, although they can help users achieve understanding, but are detailed descriptions of what was recommended. The 3rd layer clarifies the recommendations and exposes the user to the underlying rationale for that recommendation. These are minimal explanations that query a user model to find out what is necessary for the user to understand a LISP concept, LISP function, or LISP-CRITIC rule. The highest layer of information provides users a fallback capability using a rich hypertext information space in they are free to explore details or examine concepts which they still do not understand.

References

[1] Bruce K. Britton, John B. Black (editors).
Understanding Expository Text.
Lawrence Erlbaum Associates, London, 1985.

[2] T.A. van Dijk, W. Kintsch.
Strategies of Discourse Comprehension.
Academic Press, New York, 1983.

[3] G.W. Ernst, A. Newell.
ACM Monograph Series: GPS: A Case Study in Generality and Problem Solving.
Academic Press, London - New York, 1969.

[4] G. Fischer.
A Critic for LISP.
In J. McDermott (editor), *Proceedings of the 10th International Joint Conference on Artificial Intelligence (Milan, Italy)*, pages 177-184. Morgan Kaufmann Publishers, Los Altos, CA, August, 1987.

[5] G. Fischer, S.A. Weyer, W.P. Jones, A.C. Kay, W. Kintsch, R.H. Trigg.
A Critical Assessment of Hypertext Systems.
In *Human Factors in Computing Systems, CHI'88 Conference Proceedings (Washington, D.C.)*, pages 223-227. ACM, New York, May, 1988.

[6] G. Fischer, A.C. Lemke, T. Mastaglio, A. Morch.
Using Critics to Empower Users.
In *Human Factors in Computing Systems, CHI'90 Conference Proceedings (Seattle, WA).* ACM, New York, April, 1990.

[7] G. Fischer, T. Mastaglio.
Computer-Based Critics.
In *Proceedings of the 22nd Annual Hawaii Conference on System Sciences, Vol. III: Decision Support and Knowledge Based Systems Track*, pages 427-436. IEEE Computer Society, January, 1989.

[8] G. Fischer, T. Mastaglio, B.N. Reeves, J. Rieman.
Minimalist Explanations in Knowledge-based Systems.
In *Proceedings of the Twenty-Third Annual Hawaii Conference on System Sciences, Vol. III: Decision Support and Knowledge Based Systems Track*, pages 309-317. IEEE Computer Society, January, 1990.

[9] R. Flesch.
The Art of Readable Writing.
Harper & Brothers, New York, 1949.

[10] J. Frank, P. Lynn T. Mastaglio.
Using A Critic Methodology as a Computer-aided Learning Paradigm: extending the concepts. 1987.
Final Project Report for CS659 - Fall Term 1987.

[11] W. Kintsch.
The Representation of Knowledge and the Use of Knowledge in Discourse Comprehension.
Language Processing in Social Context.
North Holland, Amsterdam, 1989, pages 185-209.
also published as Technical Report No. 152, Institute of Cognitive Science, University of Colorado, Boulder, CO.

[12] J. Moore.
A Reactive Approach to Explanation.
Technical Report, USC/Information Sciences Institute, 1988.

[13] B. Reeves.
Finding and Choosing the Right Object in a Large Hardware Store -- An Empirical Study of Cooperative Problem Solving among Humans.
Technical Report, Department of Computer Science, University of Colorado, Boulder, CO, 1990. forthcoming.

[14] R.G. Schank.
Explanation Patterns: Understanding MEchanically and Creatively.
Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[15] W. Strunk, E.B. White.
The Elements of Style.
Harcourt-Brace?, New York, 1957.

[16] J.W. Wallis, E.H. Shortliffe.
Customized Explanations Using Causal Knowledge.
Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project.
Addison-Wesley Publishing Company, Reading, MA, 1984, pages 371-388, Chapter 20".

Improving Explanatory Competence[1]

Kenneth S. Murray
murray@cs.utexas.edu
Department of Computer Sciences
University of Texas at Austin
Austin, Texas 78712

Abstract

Explanation plays an important role in acquiring knowledge, solving problems, and establishing the credibility of conclusions. One approach to gaining explanatory competence is to acquire proofs of the domain inference rules used during problem solving. Acquiring proofs enables a system to strengthen an imperfect theory by connecting unexplained rules to the underlying principles and tacit assumptions that justify their use. This paper formalizes the task of improving explanatory competence through acquiring proofs of domain inference rules and describes KI, a knowledge acquisition tool that discovers proofs of rules as it integrates new information into a knowledge base. KI's learning method includes techniques for controlling the search for proofs and evaluating multiple explanations of a proposition to determine when they can be transformed into proofs of domain inference rules.

1. Introduction

Knowledge-based systems must be capable of explaining their conclusions. One approach to gaining explanatory competence is to acquire proofs of the domain inference rules used during problem solving. Possessing proofs of rules used during problem solving has several advantages. First, a rule's proofs identify support for the rule, that is, the domain principles that justify the rule's correctness. Second, a rule's proofs explicate the tacit assumptions being made when the rule is used. By identifying the underlying principles and assumptions, proofs of inference rules enable the system to justify and qualify its conclusions to the user [SWAR83], guide knowledge refinement [SMIT85], and, in the case of default reasoning when assumptions are not met, improve problem solving [STAL77]. This paper formalizes the task of improving explanatory competence through acquiring proofs and describes a system that discovers proofs of domain inference rules as it integrates new information into a knowledge base.

Acquiring proofs strengthens an imperfect theory when new information enables proving previously unsupported rules. Initially, a knowledge base (or person) often includes tentative, default rules such as "birds can fly" or "leaves are green." However, as the knowledge base is extended, and competence in the domain improves, these default rules may be annotated with deeper causal support when explanations of the rules are discovered. Gagne [GAGN85] illustrates this behavior in people with the following example:

> A student is told *In vitro experiments show that Vitamin C increases the formation of white blood cells*. The student has prior knowledge that white blood cells destroy viruses, and intuitively knows that Vitamin C is taken to fight colds, which are caused by viruses. The student realizes that Vitamin C is capable of fighting colds *because* it stimulates the creation of white blood cells, which subsequently kill cold-causing viruses.

The student identified a causal explanation of an existing belief that was neither stated in the new information nor previously known. Having discovered this explanation the student possesses greater insight into why Vitamin C is taken to fight colds. For example, the student could now explain why Vitamin C is not taken in response to similar symptoms having causes unrelated to viruses (e.g., allergies).

[1] Support for this research is provided by the Air Force Human Resources Laboratory under RICIS grant ET. 14.

This paper describes research aimed at modeling this learning behavior. In Section 2, the learning task of proof acquisition is formulated, and a terminating, although incomplete, method of performing this task is developed in Section 3. Section 4 describes KI, a knowledge acquisition tool that implements this method to discover proofs of domain inference rules as it integrates new information into a knowledge base.

2. The Learning Task

Proofs of an inference rule identify sentences in a theory that ensure its truth. However, the contents of logical proofs are not restricted to sentences having explanatory (e.g., causal) significance. For example, a proof of a rule could include an arbitrary number of trivial tautologies (e.g., $p \Rightarrow p$) not relevant to the truth of the rule. Therefore, a distinction must be made between the set of all logical proofs and proofs acceptable as explanations. Let e be a predicate on proofs such that $e(p)$ is satisfied exactly when proof p is acceptable as an explanation. For example, e might restrict proof steps to be applications of modus ponens to a particular subset of rules. Let R be a domain theory and r_i be a sentence in R. The notation p_e denotes a proof that satisfies e, and $(R - r_i) \vdash_{pe} r_i$ denotes proof p_e is a derivation of r_i from the theory $(R - r_i)$.

Each proof of a rule identifies a set of underlying principles and assumptions that justify the rule's use; different proofs may elucidate different principles and assumptions. Therefore, the goal of acquiring proofs of rules for explanation includes identifying every p_e for each rule rather than any single p_e. This learning task may be characterized as the following information processing task:

Given: a rule set R
a predicate e on proofs

Find: for each $r_i \in R$
the proof set $P_i = \{p_e \mid (R - r_i) \vdash_{pe} r_i\}$

Unfortunately, when the language used to encode rules is as expressive as first order logic (FOL) this task is not solvable. However, this task becomes decidable when the following restrictions are adopted:

Let R_e be the subset of R including only rules that may be represented as horn clauses without functions.

1) U_e, the universe of discourse for R_e, is a finite subset of U, the universe of discourse for R.

2) e admits only proofs containing non-cyclic applications of rules in R_e.

These restrictions enable the existence of a terminating method by sacrificing completeness. Intuitively, decidability is achieved by restricting the universe of discourse to a finite set. A theory equivalent to R_e can be constructed by replacing each $r_i \in R_e$ with the set of ground implications that includes every possible binding of the variables in r_i to elements of U_e. The finiteness of U_e makes such a construction possible. The result is a propositional theory and is therefore decidable.

The restricted learning task may be characterized as:

Given: a rule set R
a predicate e on proofs

Find: for each $r_i \in R$
the proof set $P_i = \{p_e \mid (R_e - r_i) \vdash_{pe} r_i\}$

While this task is guaranteed to have a terminating solution, the restrictions cannot guarantee tractability. In practice, solutions cannot be expected to discover all proofs enabled by the restricted theory. Therefore, solutions must include some mechanism to bias their search for proofs. The next section describes a method of guiding search for restricted proof acquisition.

3. Discovering Proofs through Hypothetical Reasoning

One approach to guiding deduction involves separating inference from the process of instantiating quantified formulae [McAl80]. Inference is then limited to computing the entailment of a small but "representative" set of ground propositions. This sections describes a *generate and test* search procedure that manipulates a set of ground propositions and implications to guide search for proofs of rules in R. This search is summarized by the following cycle:

1) generate a hypothetical context: a set of propositions over hypothetical instances of some small subset variables referenced by rules in R

2) generate all ground explanations (i.e., sequences of deductions) enabled by repeatedly applying rules in R_e to the propositions

3) determine if any resulting explanation can be generalized into a proof of some rule in R

4) extend the context with propositions over new hypothetical instances of variables referenced by rules in R; goto step 2.

To initiate the search, select a very restricted set of rules called *Training*; the search will be biased towards discovering proofs that make use of these rules.[2] Let *Context* be some small set of propositions that satisfy every sentence in *Training*. For example, if *Training* = {[*isa*(*x Person*) & *location*(*x Austin*) ⇒ *location*(*x Texas*)]}, then the proposition set {*isa*(*Person*$_1$ *Person*), *location*(*Person*$_1$ *Austin*), *location*(*Person*$_1$ *Texas*)} satisfies *Training*.

Second, generate ground explanations enabled by the propositions in *Context* and the rules in R_e. This involves computing all possible deductions by repeatedly applying rules in R_e to the propositions in *Context* (i.e., by exhaustive forward-chaining). While termination is guaranteed, exhaustive forward-chaining has the potential for exponential combinatorics. However, the rules are chaining on a very restricted set of instances, so, in practice, restrictions on proof construction will typically be sufficient to prevent intractable chaining.[3]

Finally, determine if any resulting explanation can be generalized into a proof of some rule. This is accomplished using techniques developed for explanation-based learning to generalize and compile explanations (e.g., [Moon88]). To continue the search for proofs, extend *Context* with additional hypothetical instances and add some set of propositions on these new instances. Then repeat the generation and evaluation of explanations as before.

Under this strategy, the search for proofs is controlled by the extensions made to *Context*. One way to generate hypothetical instances is to instantiate Skolem functions appearing in the rule set R. For example if *isa*(*Fred Person*) is a proposition in *Context*, and [*isa*(*x Person*) ⇒ *mother*(*x* $fn_1(x)$)] is a rule in R, *Person*$_1$ could be a new hypothetical instance and *mother*(*Fred Person*$_1$) a new proposition for *Context*. Now R_e can be extended with the ground implication [*isa*(*Fred Person*) ⇒ *mother*(*Fred Person*$_1$)]. This enables limited representation in R_e of rules from R that involve predicates on functions (e.g., Skolem functions). The next section describes a program that implements this method and illustrates it with an example.

---

[2] To model learning by discovery, *Training* can be any subset of R. Alternatively, when existing knowledge is being extended with new information, it is natural to prefer discovering explanations enabled by the new information. In this context, *Training* is the set of axioms being added to R.

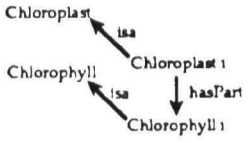
[3] To guarantee tractability, additional restrictions can be imposed on proof construction, such as a bound on the execution time allotted to compute explanations.

Figure 1: New information and the initial context

1a) New Information Provided to KI

[∀ x isa(x Chloroplast)
⇒
∃ y isa(y Chlorophyll) & hasPart(x y)]

1b) Propositions satisfying the new information



4. Acquiring Proofs During Knowledge Integration

KI is an interactive knowledge-acquisition tool being developed to help knowledge engineers integrate new information into the Botany Knowledge Base [MURR88, PORT88]. This knowledge base currently contains over four thousand frames representing plant anatomy, physiology, and development; it has been constructed in collaboration with MCC's CYC project [LENA89].

This section describes an implemented example of KI extending the Botany Knowledge Base with new information relating chloroplasts and chlorophyll. The knowledge base already contains extensive partonomic knowledge of plants and some knowledge of photosynthetic pigments, such as chlorophyll. A knowledge engineer wishes to extend the knowledge base to represent the fact that chlorophyll is a constituent part of chloroplasts (see Figure 1a). The task of KI is, in general, to identify interesting consequences of this new information and, in particular, to identify how this new information can explain existing beliefs. KI's model of knowledge integration comprises three prominent activities:

1) Recognition: identifying the knowledge relevant to new information
2) Elaboration: applying the expectations provided by relevant knowledge to determine the consequences of the new information
3) Adaptation: modifying the knowledge base to accommodate the elaborated information

4.1 Recognition

During recognition KI identifies concepts in the knowledge base that are relevant to the new information. This involves maintaining a *learning context* – a set of propositions about hypothetical instances of concepts deemed relevant to the new information. When presented with new information, KI initializes the context with propositions that satisfy the new rules (e.g., Figure 1b). To extend the learning context, KI uses *views* to determine which concepts in the knowledge base, beyond those explicitly referenced in the context, are relevant.

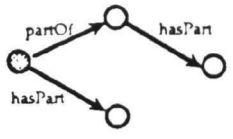
Views are sets of propositions that interact in some significant way and should therefore be considered together. Views are created by applying a generic *view type* to a domain concept. Each view type is a parameterized semantic net, represented as a set of paths emanating from a root node and used during knowledge integration as a reminding schema. Applying a view type to a concept involves binding the concept to the root node and instantiating each path. Figures 2a and b present an example view type and the view created by applying it to chloroplast.

To extend the learning context, KI identifies the views defined for concepts already contained in the learning context. Each candidate view is scored with a heuristic measure of relevance: the percentage of concepts contained in the view that are also contained in the learning context. KI presents the list of candidate views, ordered by their relevance score, to the knowledge engineer, who selects one for use.[4] The set of propositions contained in the selected view are added to the learning context. This results in a learning context comprising those concepts in the knowledge base considered most relevant to the new information.

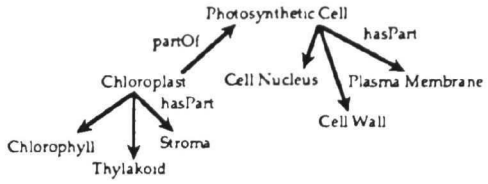
[4] Alternatively, an autonomous version of KI selects the view having the highest relevance score.

Figure 2: An example view and view type

2a) Qua Component



2b) Chloroplast Qua Component



2c) [∀ s isa(s Chloroplast) ⇒ ∃ tuvwxyz isa(t Chlorophyll) & hasPart(s t) & isa(u Thylakoid) & hasPart(s u) & isa(v Stroma) & hasPart(s v) & isa(w Photosynthetic-cell) & partOf(s w) & isa(x Cell-nucleus) & hasPart(w x) & isa(y Cell-wall) & hasPart(w y) & isa(z Plasma-membrane) & hasPart(w z)]

View type *Qua Component* identifies two paths emanating from a concept relevant to its role as a part of a physical structure (the shaded node designates the root concept). Applying this view type to chloroplast identifies the segment of the knowledge base representing chloroplast as a part of a photosynthetic cell. The path variables may have multiple bindings (e.g., the chloroplast parts include chlorophyll, stroma, and thylakoid). 2c) expresses in FOL the inference this view supports.

In addition to adding propositions contained in the selected view to the learning context, KI adds the implications that characterize the conditions under which these propositions are assumed to be true. Whenever the preconditions of a view are satisfied, the propositions contained in the view are assumed to hold. For example, when the proposition $partOf(chloroplast_1\ photosynthetic\text{-}cell_1)$ is added, the following implications are also added:

1) $isa(chloroplast_1\ chloroplast) \Rightarrow partOf(chloroplast_1\ photosynthetic\text{-}cell_1)$
2) $isa(photosynthetic\text{-}cell_1\ photosynthetic\text{-}cell) \Rightarrow partOf(chloroplast_1\ photosynthetic\text{-}cell_1)$

The first implication follows from view *ChloroplastQuaComponent* (see Figure 2c); the second follows from *PhotosyntheticCellQuaStructure* (see Rule 6 of Figure 6). Since there is high overlap among views, many such implications are added to the learning context. This enables limited representation in R_e of rules from R that involve predicates on functions (e.g., Skolem functions). The use of these implications is often essential for completing proofs of domain inference rules.

4.2 Elaboration

During elaboration KI determines how the new information interacts with the existing knowledge within the learning context. Rules in R_e are allowed to exhaustively forward-chain, propagating the consequences of the training throughout the context. For example, one consequence of chloroplasts having chlorophyll is that their color is green. Some of the domain inference rules applicable to this example are listed in Figure 3a, and the resulting conclusions are presented in Figure 3b.

KI enters a cycle of recognition (i.e., selecting views) and elaboration (i.e., applying inference rules) that explicates the consequences of the training while searching for new proofs of rules. This cycle continues until the user intervenes or the relevance scores of all candidate views fall below a progressive threshold. Figure 4 illustrates the second round of this cycle. The recognition phase extends the context of Figure 3b with the set of propositions relevant to a photosynthetic cell in its role as a producer during cell photosynthesis. The elaboration phase propagates the consequences of the new information throughout the extended context.

4.3 Adaptation

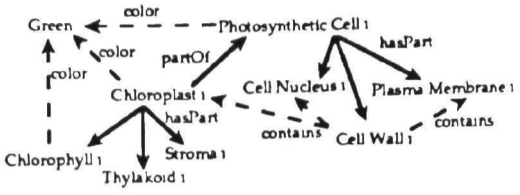
During adaptation KI determines if elaboration has revealed any new proofs of inference rules. An interesting prerequisite of discovering proofs of rules is that multiple ground explanations for some proposition must exist. When this occurs, KI determines whether any explanation can be generalized into a proof of some inference rule.

Figure 3: Example rules and inferences

3a) Example Inference Rules

1. Rule: ©partOf-Preserves-Color
   Color may be inferred from the color of the parts
   [∀ xyz partOf(x y) & color(x z) ⇒ color(y z)]
2. Rule: ©Chlorophyll_color
   Inheritance rule: Chlorophyll is inherently green
   [∀ x isa(x Chlorophyll) ⇒ color(x Green)]
3. Rule: ©Leaf_color
   Inheritance rule: Leaves are assumed to be green
   [∀ x isa(x Leaf) ⇒ color(x Green)]

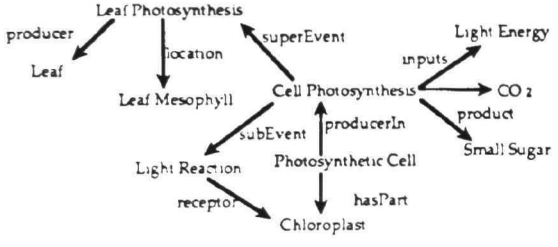
3b) The Elaborated Context



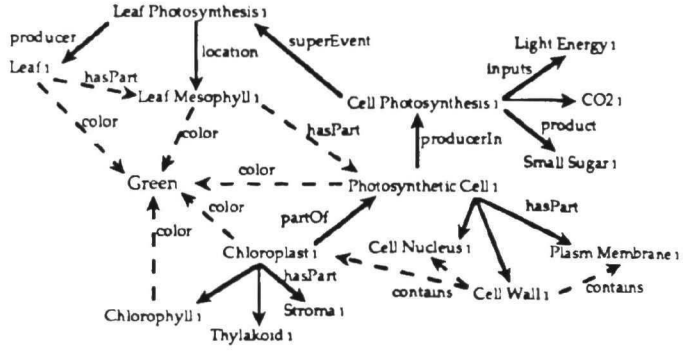
The dashed arrows indicate propositions inferred during elaboration; subscripts denote category instances (e.g., *isa(Chloroplast$_1$ Chloroplast)*).

Figure 4: Recognition and elaboration during cycle 2

4a) Photosynthetic Cell Qua Producer



4b) The Elaborated Context



Let E be the set of explanations of some proposition, and let r_i be the last rule applied in some explanation $e_i \in E$. KI evaluates each alternative explanation in E to determine if it can be transformed into a proof of r_i. First KI uses explanation-based generalization to compute the maximal generalization of each $e_j \in E\text{-}e_i$. Let ge_j be the generalization of explanation e_j. Then KI compares the consequence of ge_j to the consequence of r_i. When the consequence of ge_j is equivalent to (or subsumes) the consequence of r_i, KI searches ge_j for sub-explanations whose weakest preconditions entail the consequence of ge_j and are equivalent to (or subsume) the preconditions of r_i; each such sub-explanation constitutes a proof of r_i.

In the example, KI's elaboration of the chloroplast training produces several explanations of the proposition *color(Leaf$_1$ Green)*, two of which are presented in Figure 5. Explanation e_1 involves a single application of rule r_3 whose precondition is *isa(x Leaf)*. Since this is identical to the preconditions of the generalization of explanation e_2, the generalization of e_2 is a new proof of ©*Leaf_color*. Note the importance of implications that explain propositions arising from views. As shown in Figure 6, the views *LeafQuaStructure*, *MesophyllQuaStructure*, *PhotosyntheticCellQuaStructure* and *ChloroplastQuaStructure* all chain together to demonstrate that leaves contain chlorophyll.

This example shows how KI discovers a proof for the existing rule ***leaves are green*** while integrating the new information ***chloroplasts have chlorophyll.*** This proof improves the system's explanatory competence by revealing the tacit assumptions (e.g., ***mesophyll contains photosynthetic cells***) and domain principles (e.g., ***an object's color is determined by the color of it's parts***) that justify the rule's use. For example, the proof provides an answer the query ***why are leaves green?***

Figure 5: Two explanations of $color(leaf_1\ Green)$

explanation e_1

$color(Leaf_1\ Green)$
$\Leftarrow_{r3}\ isa(Leaf_1\ Leaf)$

explanation e_2

$color(Leaf_1\ Green)$
$\Leftarrow_{r1}\ partOf(Mesophyll_1\ Leaf_1)$
$\Leftarrow_{r4}\ isa(Leaf_1\ Leaf)$
$color(Mesophyll_1\ Green)$
$\Leftarrow_{r1}\ partOf(Photosynthetic\text{-}cell_1\ Mesophyll_1)$
$\Leftarrow_{r5}\ isa(Mesophyll_1\ Mesophyll)$
$\Leftarrow_{r4}\ isa(Leaf_1\ Leaf)$
$color(Photosynthetic\text{-}cell_1\ Green)$
$\Leftarrow_{r1}\ partOf(Chloroplast_1\ Photosynthetic\text{-}cell_1)$
$\Leftarrow_{r6}\ isa(Photosynthetic\text{-}cell_1\ Photosynthetic\text{-}cell)$
$\Leftarrow_{r5}\ isa(Mesophyll_1\ Mesophyll)$
$\Leftarrow_{r4}\ isa(Leaf_1\ Leaf)$
$color(Chloroplast_1\ Green)$
$\Leftarrow_{r1}\ partOf(Chlorophyll_1\ Chloroplast_1)$
$\Leftarrow_{r7}\ isa(Chloroplast_1\ Chloroplast)$
$\Leftarrow_{r6}\ isa(Photosynthetic\text{-}cell_1\ Photosynthetic\text{-}cell)$
$\Leftarrow_{r5}\ isa(Mesophyll_1\ Mesophyll)$
$\Leftarrow_{r4}\ isa(Leaf_1\ Leaf)$
$color(Chlorophyll_1\ Green)$
$\Leftarrow_{r2}\ isa(Chlorophyll_1\ Chlorophyll)$
$\Leftarrow_{r7}\ isa(Chloroplast_1\ Chloroplast)$
$\Leftarrow_{r6}\ isa(Photosynthetic\text{-}cell_1\ Photosynthetic\text{-}cell)$
$\Leftarrow_{r5}\ isa(Mesophyll_1\ Mesophyll)$
$\Leftarrow_{r4}\ isa(Leaf_1\ Leaf)$

The notation $p \Leftarrow_i q$ denotes p is infered from q by rule i (see Figures 3 and 6).

Figure 6: Inferences enabled by views and required for the proof

Rule 4) *LeafQuaStructure*
[∀ w isa(w Leaf) ⇒ (∃ xyz isa(x Mesophyll) & partOf(x w) & isa(y Epidermis) & partOf(y w)
& isa(z Vascular-network) & partOf(z w))]

Rule 5) *MesophyllQuaStructure*
[∀ x isa(x Mesophyll) ⇒ (∃ y isa(x Photosynthetic-cell) & partOf(y x))]

Rule 6) *PhotosyntheticCellQuaStructure*
[∀ v isa(v Photosynthetic-cell) ⇒ (∃ wxyz isa(w Cell-nucleus) & partOf(w v) & isa(x Cell-wall) & partOf(x v)
& isa(y Chloroplast) & partOf(y v) & isa(z Plasma-membrane)
& partOf(z v))]

Rule 7) *ChloroplastQuaStructure*
[∀ w isa(w Chloroplast) ⇒ (∃ xyz isa(x Chlorophyll) & partOf(x w) & isa(y Thylakoid) & partOf(y w)
& isa(z Stroma) & partOf(z w))]

(e.g., *leaves are green because they contain chlorophyll*). Alternatively, the proof guides dependency-directed backchaining to identify assumptions that explain why a particular leaf is not green (e.g., *the leaf's mesophyll has no photosynthetic cells*).

4.4 Strengths, Limitations, and Future Work

KI's approach to knowledge integration involves creating a *hypothetical model* comprising concepts relevant to the new information, and then using the model to derive the consequences of the new information for concepts represented in the model. Reasoning with a single, propositional model (e.g., a model of a hypothetical leaf), rather than reasoning about entire classes of objects (e.g., models of all possible leaves) provides greater focus and tractability. However, this prevents KI from discovering many proofs that alternative models would reveal. Furthermore, KI is currently not capable of exploring all the alternative, and often mutually-inconsistent, behaviors of a model that frequently arise during qualitative simulations [KUIP87]. This prevents KI from discovering

many proofs that a single model may be capable of revealing under varying assumptions. Future work should develop methods for guiding the exploration of alternative models and the *possible worlds* for a single model.

The inferences completed with the model are not explicitly selected: rules exhaustively forward-chain. This type of reasoning corresponds to what Johnson-Laird calls *implicit inference* – the automatic, seemingly effortless inferences humans make during mundane tasks, such as discourse comprehension [JOHN83]. The complement of implicit inference is *explicit inference* – the intentional and conscious reasoning humans perform during problem solving. Currently, KI is not capable of demonstrating this kind of goal-directed elaboration. Future research must address developing methods for interleaving these two types of inference.

5. Summary

Explanation plays an important role in a system's ability to acquire knowledge, solve problems, and establish the credibility of its conclusions. One approach to gaining explanatory competence is acquiring proofs of the inference rules used during problem solving. Acquiring proofs enables a system to strengthen an imperfect theory as previously unexplained rules are connected to the underlying principles and tacit assumptions that justify their use.

KI is a knowledge acquisition tool that strengthens an existing domain theory by discovering new proofs of inference rules. When new information is provided, KI actively searches for proofs of existing beliefs that are enabled by the new information. This requires methods for restricting both the universe of discourse and the use of inference rules that include predicates on functions.

KI exploits a type of domain knowledge called *views* to precisely manage a context comprising ground propositions used during the search for proofs. Views are knowledge-base segments composed of interrelated propositions that should be considered collectively. Each view embodies the use of functions to create entities over which propositions are asserted. Separating the use of functions to create entities from the problem of proving theorems enables KI to guide its search for proofs of domain inference rules.

References

[GAGN85] Gagne, E.D. *The Cognitive Psychology of School Learning*, Boston: Little, Brown and Company, 1985.

[JOHN83] Johnson-Laird, P.N. *Mental Models*, Harvard University Press, 1983.

[KUIP87] Kuipers, B.J., and Chiu, C. Taming Intractable Branching in Qualitative Simulation. *Proceedings of the Tenth International Joint Conference on Artificial Intelligence* (1987).

[LENA89] Lenat, D.B., and Guha, R.V. *Building Large Knowledge-Based Systems: Representation and Inference in the CYC Project*, Addison-Wesley, 1989.

[MCAL80] McAllester, D.A. An Outlook on Truth Maintenance. AI Memo No. 551, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1980.

[MOON88] Mooney, R.J. *A General Explanation-Based Learning Mechanism and Its Application to Narritive Understanding*, PhD Dissertation, Computer Science Department, University of Illinois at Urbana-Champaign, 1988.

[MURR88] Murray, K.S. KI: An Experiment in Automating Knowledge Integration. Technical Report AI-88-90, Artificial Intelligence Laboratory, Department of Computer Sciences, University of Texas at Austin, 1988.

[SMIT85] Smith, R.G., Winston, H.A., Mitchell, T.M., Buchanan, B.G. Representation and Use of Explicit Justification for Knowledge Base Refinement. *Proceedings of the Ninth International Joint Conference on Artificial Intelligence* (1985).

[STAL77] Stallman, R.M., and Sussman, G.J. Forward Reasoning and Dependency Directed Backtracking in a System for Computer-Aided Circuit Analysis. *Artificial Intelligence 9* (1977).

[SWAR83] Swartout, W.R. XPLAIN: A System for Creating and Explaining Expert Consulting Programs. *Artificial Intelligence 21* (1983).

[PORT88] Porter, B., Lester, J., Murray, K., Pittman, K., Souther, A., Acker, L., and Jones, T. AI Research in the Context of a Multifunctional Knowledge Base: The Botany Knowledge Base Project. Technical Report AI-88-88, Department of Computer Sciences, University of Texas at Austin, 1988.

Incremental Envisioning: The Flexible Use of Multiple Representations in Complex Problem Solving

Malcolm I. Bauer Brian J. Reiser

Cognitive Science Laboratory
Princeton University

Abstract: In this paper we describe two properties of most psychological and AI models of scientific problem solving: they are one-pass, and feedforward. We then discuss the results of an experiment which suggests that experts use problem solving representations more flexibly than these models suggest. We introduce the concept of *incremental envisioning* to account for this flexible behavior. Finally, we discuss the implications of this work for psychological models of scientific problem solving and for AI programs which solve problems in scientific domains.

1 Introduction

Complex problem solving often involves the use of several representations. For example, when problem solving in scientific domains, people often use formal mathematical equations, many types of diagrams, and informal conceptual intuitions in the course of reasoning. In this paper, we consider people how coordinate the use of several representations while solving a problem in a scientific domain.

2 The Traditional View

In his book, *How to Solve It*, Polya (1945) described what he believed were the four phases of problem solving: 1) understanding the problem 2) developing a plan to solve it 3) carrying out the plan, and 4) checking over the solution. Most psychological models of mathematics and science problem solving (e.g., Larkin, McDermott, Simon, & Simon, 1980; Chi, Feltovich, & Glaser, 1981; Riley, Greeno, & Heller, 1983) and programs in AI which solve problems in science (de Kleer, 1975; Skorstad & Forbus, 1989) may be considered instantiations of this view within specific domains. For example, Larkin et al. (1980) propose that physics experts solving mechanics problems will begin by sketching a picture of the described problem and selecting a set of principles. From these they construct a representation of the problem containing relevant physical entities (i.e. understanding the problem). This conceptual representation of the problem is then re-represented as a set of physics equations (constructing a plan). Finally the equations are solved algebraically (carrying out the plan). In AI, de Kleer's (1975) Newton program solves simple kinematics problems in a similar manner. Given a description of a physical scenario and a question about that scenario (e.g. "How fast will the block be traveling at point A?"), Newton first constructs an *envisionment* or representation of what will happen in the scenario. Second, it constructs a general plan of what must be done to solve the problem. Finally it accesses the relevant mathematical knowledge and solves associated equations until the desired quantity is found.

In both models, as in most subsequent models of scientific problem solving, problem solving is accomplished by proceeding sequentially through the phases described by Polya. First some sort of general conceptual model of the problem situation is constructed (in the case of Newton, this conceptual model is called an envisionment). Second, a general plan for solving the problem is constructed, embodied either as the general principles operating or as the crucial states that must be solved for. Fi-

nally, the plan cues the relevant equations, which are then solved. Polya's fourth phase, checking over the solution, is rarely included. To summarize, these models can be characterized as comprised of some or all of the following phases.

1. Construct a conceptual understanding of what is occurring in the problem.

2. Develop a plan to solve the problem. Typically this involves deciding what principles are relevant and which equations will be used.

3. Construct and solve the relevant equations.

4. Check the solution.

The models have two key properties. First, the models are *feed-forward* because information flows from earlier phases to later phases, but never in the other direction. For example, inferences made while solving equations are never passed back to earlier phases to enhance conceptual understanding.

Second, the models are *one-pass serial.* All processing within each phase is completed before the next phase is initiated. There is a block of processing using to understand the problem conceptually followed by a planning phase, which is followed by a solving phase. This property is distinct from the first one in that it is possible to have an iterative model rather than a one-pass model which would still be feed-forward. Such a model would loop through the phases several times but only passing information "forward," for example, never using phase 3 inferences in later phase 1 processing.

For fairly simple problems, there is psychological evidence that expert problem solving fits this one-pass, feed-forward model (Larkin et al., 1980). Similarly, in AI, this model has been used to solve several types of physics problems (Skorstad & Forbus, 1989; de Kleer, 1975). However, there are several reasons to believe this model does not completely characterize how experts solve all types of problems. Similarly, there are reasons to believe that this kind of AI architecture will not be able to solve many kinds of problems.

First, for complex problems, short-term memory restrictions may require people to cycle through the phases, solving pieces of the problem each time, to put together a coherent complete solution, rather than doing all the required reasoning in each phase before initiating the next phase.

Second, memory considerations aside, for difficult problems, experts may need to use several kinds of representations simultaneously to characterize a problem conceptually. This may include particular equations, theoretical models from physics, and commonsense intuitions. Roschelle and Greeno (1987) give anecdotal evidence to support this in protocols where expert physicists use both Newtonian physics models and commonsense intuitions about a physical situation to how objects will behave.

Third, de Kleer (1975) describes a class of problems he terms *indefinite* that his program is unable to solve. He claims it can't solve these problems because the program lacks flexibility. It needs to access information from different phases of its problem solving, but cannot because it is a one-pass feed-forward model. For example, certain problems may require some calculations be performed (phase 3) in order to complete conceptual understanding (phase 1).

Finally, recent work in qualitative reasoning (Sacks, 1988) has focused on interpreting formal symbolic solutions qualitatively. In many scientific disciplines, coming up with a formal symbolic solution to a problem (the result of phase 3) is not the final goal, as it is in the models above. Instead the goal is to understand what the solution means at a conceptual level (phase 1). The work in qualitative reasoning focuses on interpreting the results of phase 3 in terms of the conceptual representations utilized in phase 1. In one-pass feed-forward models, this is impossible as passing back results from phase 3 to phase 1 does not occur.

3 An Empirical Investigation of Multiple Representations

The present research is concerned with understanding expert performance in situations that experts find more challenging. The psychological models described above were typically derived from expert performance on problems requiring little effort for the experts. For the reasons above,

we suggest that a one-pass feed-forward model will be inadequate to completely characterize expert performance on more complex problems. We examine expert performance on moderately difficult mechanics problems in physics. Expert performance on "easy" problems has been studied extensively in mechanics so this provides a good basis for comparison. We are interested in investigating whether one-pass feed-forward models are inadequate to explain expert behavior, and if not, what is it that experts do beyond these models in those situations.

3.1 Design and Materials

We selected four hard mechanics problems. Three of the four were taken from a review text (Wells & Slusher, 1983). The fourth was created by one of us. Simplified versions of each of the hard problems were constructed. These used the same principles necessary to solve the hard problems, but the physical scenarios in which those principles had to be used were greatly simplified. Examples are shown in Figure 1.

Subjects were graduate students drawn from the Mechanical Engineering and Physics departments at Princeton University. There were 16 subjects in total, although the analyses in this paper focus on the first 6 subjects. Each subject solved four problems, two easy and two hard. No subject was given a hard problem and its corresponding easy problem. Subjects were asked to "think out loud" while solving the problems. The sessions were videotaped. We transcribed all subjects' actions which included verbal statements, writing an equation, drawing a diagram, modifying an equation or diagram, and pointing to an equation or diagram. Because we wished to examine the transitions among the kinds of representations used, we coded protocol statements according to the kind of information used and the type of action being performed. Our analyses consider only the *information heeded* by the subject, rather than attempting to categorize the actual processes which are acting upon that information (Ericsson & Simon, 1984). Recognizing the transitions was also facilitated by the fact that, in addition to the verbal protocols, the transcripts also contained all cases where subjects modified or pointed to an equation or diagram. Protocol statements were classified into one of the eight basic categories described below:

Categorization: Subject states a category to which the problem belongs.

Rehearsal: Subject reads or re-reads problem, or restates a fact previously found.

Physical Reasoning: Subject identifies a particular physical quantity in problem, or states what occurs in the scenario, without the use of equations.

Diagram Use: Subject draws, labels, or points to a diagram.

Miscellaneous: This category includes explicit statements involving planning, and stating basic physics principles.

Mapping: Subject explicitly maps information from one representation to another.

Formal Symbolic Manipulation: Subject recalls, writes down, or performs any operation on an equation.

Qualitative Mathematical Reasoning: Subject considers an equation and reasons about it qualitatively.

Setting Goals, Hitting Impasses: Subject states a goal, or makes a statement that he or she has hit an impasse (e.g. "I'm stuck.")

Each general category was divided in several subcatgories to code the kind of action performed, if one was explicit. For example, Diagram Use had three subcategories: writing down part or all of a diagram, pointing to a diagram, and labeling part of a diagram with an equation or symbol.

3.2 Analyses

First, subjects indeed found the hard problems more difficult that the easy problems. On average it took the subjects 4.3 minutes to solve the easy problems and 18 minutes to solve the hard problems. In terms of the coding scheme, transcripts for the easy problems contained an average of 38.3 steps and for the hard problems, 122.3. All of the easy problems were solved correctly, but only 75% of the hard problems.

5.13. In the turntable arrangement shown in Fig. 5-13, block A has a mass of 0.9 kg, block B has a mass of 1.7 kg, and the blocks are 13 cm from the axis of rotation. The coefficient of static friction between the blocks, and between the blocks and the turntable, is $\mu_s = 0.1$. Consider the friction and the mass of the pulley in Fig. 5-13(a) as negligible. Find the angular speed of rotation of the turntable for which the blocks just begin to slide.

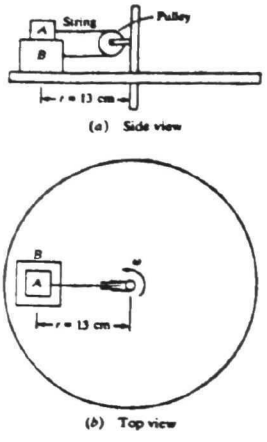


Fig. 5-13

A 4 kg block rests 0.6 meters from the center of a turntable. If the coefficient of static friction between the block and the turntable is .2, find the maximum angular velocity of the turntable for which the block will not slide.

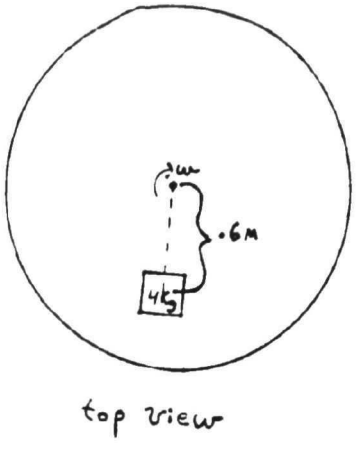


Figure 1: A hard problem (left) and its corresponding easy problem (right)

There are several analyses which can be used to evaluate the types of problem solving sequences. As a first step towards investigating these reasoning events, we divided each protocol into quarters according to the total number of codes in the protocol, and plotted the average percentage of each category of action within a quartile. The graph in Figure 2 is a quartile plot for the most prevalent codes.

The most dramatic effect is the rise in the percentage of formal symbolic manipulation, starting at 13% in the first quartile and rising fairly linearly to 60% in the fourth quartile. In addition, the physical, diagrammatic, and rehearsal codes start at around 20% in the first quartile, and slowly drop until they are all about 4% in the last quartile. This overall trend of the increase in formal manipulation and decrease in actions of with conceptual understanding is generally in keeping with the one-pass feed-forward model. However, the fact that there is even 15% formal manipulation in the first quartile and some actions of conceptual understanding in the last quartiles suggests that the one-pass feed-forward model of phases does not tell the whole story. In fact, some solving of equations (associated with phase 3) occurred before physical reasoning (associated with phase 1) thus the strict ordering of the phases is not being followed completely (the "one-pass" property).

Another way to evaluate the model is to examine the number of transitions between the formal and conceptual representations. A one-pass model would predict very few. Transitions would occur only when creating the formal equations from the conceptual representation. For the easy problems the average number of transitions was 4, while for the hard, 14. This again is evidence which supports the claim that subjects are not strictly "one-pass." Looking at the step/transition ratio, we find that this is approximately 1 transition for every 9 steps, which seems to be many more than would be expected if problem solving occurs as a long episode of conceptual work followed by an episode of planning, followed by an episode of formal symbol manipulation. Instead, it appears that subjects shift between phase 1 and phase 3 relatively frequently.

There is also evidence that subjects were not strictly feed-forward (property 1) while solving these problems as well. In many instances subjects

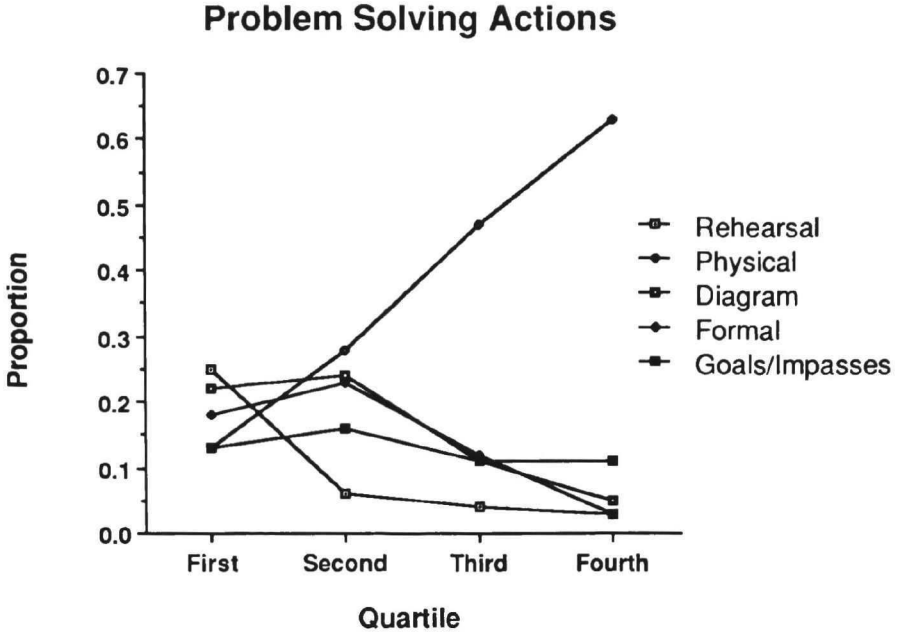


Figure 2: Quartile Plot of Problem Solving Actions

actually interpret derived equations to enrich their conceptual understanding in the course a problem. This *backward mapping* violates the feed-forward property described above. A typical example of this type of episode occurred when one subject was solving a problem about a falling rope:

> "Here's an old equation: $V^2 = {V_0}^2 + 2as$ [subject writes equation down] This is 0 [subject crosses out V_0 leaving the equation $V^2 = 2as$], so the velocity as it hits the table [points to rope in diagram] is gonna be a function of how far away from the table it was."

Here the subject has recalled an equation, applied some known quantitative information ($V_0 = 0$) and then interpreted the meaning of the expression conceptually, updating his conceptual understanding of the problem. The subject has done some work which would be classified as phase 3, but then applied it to work which would be classified as phase 1. For hard problems, this type of backward mapping occurred an average of 3.6 times per problem. This type of shift cannot occur in any model which is purely feed-forward. A more flexible model of reasoning is required.

4 Discussion

We have presented evidence that suggests that often people do not completely follow the one-pass feed-forward model. Their behavior is not strictly one-pass: they shift between understanding (phase 1) and solving (phase 3) many times in the course of reasoning about a problem. Similarly, their behavior is not strictly feed-forward: often they will use the results of solving aspects of a problem (phase 3) to enhance their conceptual understanding of the problem (phase 1) which will in turn enable them to solve other parts of the problem. Interestingly, there are different representations associated with each phase. Phase 1 is associated with building a conceptual under-

standing of a problem. In previous models, conceptual understanding is represented as a mental model (Roschelle & Greeno, 1987; Hegarty, Just, & Morrison, 1988), or an envisionment of what could happen (de Kleer, 1975; Roschelle & Greeno,1987). The representation used in the solution phase (phase 3) is a set of formal symbolic equations (Larkin et al, 1980). In this section, we discuss how experts coordinate these representations in problem-solving.

In most models of scientific problem solving, all of the understanding phase occurs at the beginning. Experts read the problem description, construct a representation of what's going on, and then set about to solve the problem. Instead, we propose that experts perform *incremental envisioning*: they successively refine their conceptual understanding of a problem as they work through it. There are two general ways in which subjects shift from one representation to the other. The first is a shift from envisioning to a representation associated with another phase. The second is a shift from working on the equations back to increasing the conceptual understanding of a problem.

We propose that the first kind of shift occurs largely to reduce the load of working memory. Here subjects will be thinking about what's going on in a problem and discover something relevant to one of the other phases – either an equation will be cued, an important subgoal discovered, or an important physical insight gained. The subject will then stop developing their conceptual understanding and shift to preserve that relevant piece of information, either by writing down the equation and doing some formal symbolic manipulation, by adding the physical insight to a diagram, or by stating clearly what that new goal is. In this type of shift, envisioning is momentarily halted and the important ramifications of it are propagated to other representations and preserved. Then envisioning is resumed. This is still feed-forward in that work in the other representations does not affect the envisioning, however, in contrast to the one-pass models described earlier, the process is incremental. By propagating new relevant information to other representations, the other representations are built up as the envisionment progresses. In this way, neither the entire envisionment, nor its ramifications for the other representations, have to be held in memory all at once.

To investigate this kind of incremental envisioning more precisely, we constructed the transition table containing the probability of each kind of action directly following a physical reasoning (envisioning) event (see Table 1). It is clear from the transition table that all the physical reasoning does not occur in one block, but instead shifts to other kinds of actions quite regularly. The probability of shifting to working on diagrams, shifting to working on equations, and shifting to setting goals, are all close to, if not greater than the probability of continuing with the physical reasoning.

Categorization	0.015
Rehearsal	0.073
Physical Reasoning	0.188
Diagram Use	0.272
Miscellaneous	0.019
Mapping	0.042
Formal Symbolic Manipulation	0.226
Qualitative Mathematical Reasoning	0.004
Setting Goals, Hitting Impasses	0.153

Table 1

The following examples demonstrate these feed-forward shifts:

Physical cuing Formal:

> we have the man..of mass M ..on the ladder which is a force Mg..

Here the subject began to describe the scenario (man on ladder) and shifted to writing the symbolic expression for the weight of the man (force Mg).

Physical cuing Diagram:

> that's the force needed to keep that thing going in the circle.. [subjects adds arrow to diagram]

In this example, the subject envisioned what a particular force is going to do, and then preserved that inference by adding an arrow (drawn in a circle) to a diagram.

physical cuing goals:

> ...we remain on the same circle, but I am moving on the circle and he's not..so the

> maximum time it's going to take is the time I need to make a complete circle, and that will be the worst case...so all I know have to know is the tangential speed...

Here the subject reasoned conceptually about what will happen, and then recognized a new goal to be acheived (phase 2).

In second kind of shift, the conceptual understanding phase is resumed because of an event that occurs in one of the other representations. Often, though not always, the subject returns to phase 1 to help resolve a difficulty arising in another phase. Envisioning might be resumed because of the realization that the problem cannot be solved from the current equations, for example the subject realizes that there are too few equations for the number of unknowns. This kind of shift also may be necessary to solve indefinite problems. Often it involves interpreting the results of symbolic manipulation (phase 3), as in the rope example given in the previous section. This may be done to check the validity of the derived equation, (if the interpreted equation makes sense conceptually it's more likely to be true). Occasionally, this kind of shift occurs simply to update the conceptual understanding of the problem, not to resolve any particular difficulty. However, this may cause the subject to gain new insight into the problem. Again, the rope protocol above is an example of this. Each of these cases is an example of iterative refinement as the understanding of the problem is updated with each return to envisioning. Some examples of this second type of shift are:

Too few equations:

> that's for that equation so that's two equations two unknowns.. no wait a minute two equations four unknowns<laughs> but we have two more equations.. which are the uhhh... wait a minute, [points to problem diagram] why does l interfere here?.... uhmmm.. well let's see physically what happens? If we start at this point....

Here the subject was working on the equations when he realized there are too few equations for the number of unknowns. In trying to come up with the other two equations, the subject shifted back to thinking about the problem conceptually.

Shift after finding impossible results:

> and we don't want a negative because we have to take a square root... and I screwed up the sign somewhere here.. A minus B... [points at minuses in equations] that's a minus, minus, minus, minus 2.... b....where did I lose my sign?.. [pointing at equations] T zero minus minus minus .. did I .. mess up...my forces while on this little thing?..... [points to diagram and checks equations against diagrams] where are my forces going here?..tension.... $muMg...muMg$ that's gotta be the opposite of that one... that one going that way...

Here the subject, in the course of solving the problem came up with the square root of a negative number. In trying to track down the error, he switched back to reasoning conceptually about what happens in the physical scenario.

To summarize, incremental envisioning occurs in two main ways. The first is done to preserve the results of envisioning and involves propagating each new result to other representations. In the second, an event in phase 3 causes envisioning to be resumed often to help resolve a difficulty arising in phase 3 processing. In this case, results from phase 3 are propagated back to phase 1.

5 Conclusions

In this paper we have described a class of scientific problem solving models called one-pass feed-forward models. We then described an experiment which suggested that expert problem solving behavior involved more than could be accounted by these models. Finally, we proposed that experts perform *incremental envisioning* as a way of describing the kinds of behaviors not characterized by one-pass feed-forward models. In this section, we briefly elaborate the implications of this work for AI and psychology.

At first glance, the first type of incremental envisioning, feed-forward shifts, may not seem useful for AI programs that solve scientific problems. We suggested this first type is used to overcome short-term memory constraints in humans. However, computers have no such limitations, so there

is less need to preserve the representational ramifications of envisioning in the course of performing envisioning. Instead, the entire envisionment may be saved, and used as a whole to help in planning and solving the problem. Indeed, most scientific problem solvers work this way (de Kleer, 1975; Skorstad & Forbus, 1989). However, there are many scientific problems in which a complete envisionment need not be performed. Creating one, without reference to what must be solved for in the problem, is inefficient in these cases. Also, there are many problems for which it is impossible to construct a complete envisionment from the given information, yet the problems are solvable for the particular question being asked. In these cases, creating partial envisionments and propagating the results of envisioning during the course of envisioning is essential to deriving a solution.

The importance of the second type of incremental envisioning, backward mapping, is clear. In many cases, it may be necessary to solve part of a problem in order to complete an envisionment to solve the rest of the problem. De Kleer's indefinite problems fall into this class. Similarly, in other circumstances, it may be desirable to interpret equations conceptually. Work in AI along these lines is already being done (Sacks, 1988; see Forbus, 1988 for comprehensive review).

For psychological models, this work demonstrates that experts use representations more flexibly than has been thought. Models of expert problem solving must take into account this added flexibility. We are currently developing a computational model of incremental envisioning in problem solving. Roschelle and Greeno's (1988) relational model is a step in this direction as well.

Acknowledgments: The research reported here was supported in part by contract MDA903-87-K-0652 from the Army Research Institute, by a research grant from the James S. McDonnell Foundation, and by a grant-in-aid of research from the Sigma Xi, The Scientific Research Society. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Army or the McDonnell Foundation.

References

Chi, M. T. H., Feltovich, P. J., & Glaser, R. (1981) Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5,121–152.

de Kleer, J. (1975). *Qualitative and quantitative knowledge in classical mechanics.* Master's thesis, Department of Electrical Engineering, Massachusetts Institute of Technology, Cambridge, MA.

Ericsson, K. A., & Simon, H. A. (1984). *Protocol Analysis.* Cambridge, MA: MIT Press.

Forbus, K. D. (1988). Qualitative physics: Past, present, and future. In H. E. Shrobe (Ed.), *Exploring Artificial Intelligence*, San Mateo, CA: Morgan Kaufmann.

Larkin, J. H., McDermott, J., Simon, D. P., & Simon, H. A. (1980). Models of competence in solving physics problems. *Cognitive Science*, 4, 317–345.

Polya, G. (1945). *How to Solve It.* Princeton, NJ: Princeton University Press.

Riley, M. S., Greeon, J. G., & Heller, J. I. (1983). Development of children's problem solving ability in artihmetic. In H. P. Ginsberg (Ed.), *The development of mathematical thinking*, NY: Academic Press.

Roschelle, J., & Greeno, J. G. (1987). mental models in expert physic reasoning. *Report No. GK-2*, University of California, Berkeley.

Sacks, E. (1988). Qualitative analysis by piecewise linear approximation. *Artificial Intelligence in Engineering*, 3, 3.

Skorstad, G. & Forbus, K. (1989). Qualitative and Quantitative Reasoning about Thermodynamics. *Proceedings of the 11th Annual Conference of the Cognitive Science Society*, Ann Arbor, Michigan: Lawrence Erlbaum.

Wells, D. A., & Slusher, H. S. (1983) *Physics for Engineering and Science.* U.S.A: McGraw-Hill.

Task-Based Criteria for Judging Explanations[1]

David B. Leake
Center for Research on Concepts and Cognition, Indiana University
leake@cogsci.indiana.edu

Abstract

AI research on explanation has not seriously addressed the influence of explainer goals on explanation construction. Likewise, psychological research has tended to assume that people's choice between explanations can be understood without considering the explainer's task. We take the opposite view: that the influence of task is central to judging explanations. Explanations serve a wide range of tasks, each placing distinct requirements on what is needed in an explanation.

We identify eight main classes of reasons for explaining novel events, and show how each one imposes requirements on the information needed from an explanation. These requirements form the basis of dynamic, goal-based explanation evaluation implemented in the program ACCEPTER. We argue that goal-based evaluation of explanations offers three important advantages over static criteria: First, it gives a way for an explainer to know what to elaborate if an explanation is inadequate. Second, it allows cross-contextual use of explanations, by deciding when an explanation built in one context can be applied to another. Finally, it allows explainers to make a principled decision of when to accept incomplete explanations without further elaboration, allowing explainers to conserve processing resources, and also to deal with situations they can only partially explain.

Introduction

It seems obvious that people who explain usually do so for a reason— that explanation is done to serve an overarching task. Explanations allow people to understand unexpected situations, in order to deal with them more effectively. However, the effect of overarching tasks on explanation has received little attention in psychology and artificial intelligence. Attribution theory [Heider, 1958], the central current in psychological study of people's judgement of explanations, tries to account for their judgements in a context-independent way. AI work in explanation-based learning (EBL) also fails to address the influence of changing goals on explanation (see [DeJong, 1988] for an overview of EBL research).

Context-independent theories fail in two main ways. First, they simply cannot account for the choices that people make. Second, in an AI system, they limit system performance by sometimes failing to accept explanations that are useful, sometimes accepting explanations that are not. Context-independent theories often require complete explanations, showing

[1]This work was conducted in part at the Center for Research on Concepts and Cognition at Indiana University, supported by a grant from Indiana University, and at Yale University, supported in part by the Defense Advanced Research Projects Agency, monitored by the Office of Naval Research under contract N0014-85-K-0108 and by the Air Force Office of Scientific Research under contract F49620-88-C-0058.

necessary and sufficient conditions for occurence of the outcome being explained. However, a robbery victim who wants to keep from being robbed again does not need to form a complete picture of the robbery to benefit in the future: as long as he realizes that leaving a window open made the robber choose to rob his house, he can prevent future robberies. Nor is a complete explanation guaranteed to be useful: An account of the robber's motivation, no matter how complete, would not help the victim make his house more secure.

To develop useful criteria for evaluating explanations, we must consider how explanations are used. The following sections identify eight purposes for explanation, and illustrate how each purpose places different requirements on the information that an explanation needs to provide. These requirements in turn determine the goodness of an explanation. We illustrate the effect of purpose on explanation with output from evaluation of an explanation for two different purposes by ACCEPTER, a program that examines the information provided in explanations, according to the information requirements for a range of user-specified tasks.[2] We conclude with a discussion of how task-based evaluation allows explanation-based approaches to be applied despite the lack of a a complete explanation.

Attribution theory

Seminal work by Heider [1958] initiated psychological research into how people decide to favor certain explanations. Heider originated *attribution theory*, which investigates how people decide whether to explain an action in terms of features of its actor, or features of the environment. (Most work in attribution theory assumes that either personal or situational factors will apply, but not both.) Kelley's *covariation principle* gives a hypothesis for how people make this decision [Kelley, 1967]. It suggests that people look at covariation across different people, other entities, and time, in order to decide which type of factor applies. For example, if John dislikes a movie, but most other viewers are enthusiastic, Kelley's covariation principle suggests that John's dislike should be explained by aspects of John, rather than the movie.

The covariation principle makes no reference to how the explanation might be used. More recent work has noted one area in which attributions are influenced by overarching goals: *excuse theory* shows how people displace blame by focusing on external reasons when explaining their own bad performance (see, for example, [Mehlman and Snyder, 1983]). Excuse theory is built on Kelley's model, and shows how the desire to form excuses makes an explainer manipulate the balancing that would otherwise be determined by covariation. However, attribution theory has not addressed the role of other goals in explanation. We will show both that other goals exert strong influences, and that they require characterizing information along dimensions beyond the person–situation distinction that is central to attribution theory.

Explanation-based learning

AI research on explanation-based learning (EBL) has shown that explanations provide valuable guidance for feature selection in new situations. However, EBL research concentrates on how to learn from an explanation presented to the system, giving less attention to

[2] ACCEPTER is a system that detects gaps in its knowledge that require explanation, and that evaluates explanations' plausibility as well as the type of information they provide. Those phases of the system are beyond the scope of this paper, but are described further in [Leake, 1988].

explanation selection. To the extent that it has addressed intended use, it has concentrated on explaining for a single purpose: efficient object recognition (*e.g.*, [Mitchell *et al.*, 1986] and [Kedar-Cabelli, 1987]). Keller [1988] points out that studying EBL only in the context of object recognition has had a strong, and falsely limiting, effect on analysis of explanations: an explanation that permits effective object recognition may not be useful for other tasks.

Some AI systems do apply their explanations to other tasks (*e.g.*, plan repair in [Hammond, 1986]), but rely on being presented with good explanations. This cannot be assumed when using explanations from external sources, or re-using explanations built in other contexts. For example, if we want to find out why our brand X car will not start, in order to repair the problem, we might ask a friend. If he had previously urged us to buy a different brand, he might reply "because brand X is junk." A person would not be satisfied with this answer; nor should an EBL system.

Tasks that drive explanation

In order to understand human explanation, and to build systems that can explain effectively, we must first look at *why* explanation is done. Once we know the purposes for explanation, we can investigate what makes explanations good for those purposes— what information the explanations must provide.

We have identified eight primary tasks served by explanation, each of which imposes different requirements on what constitutes a good explanation. We sketch these tasks below, and describe the requirements they impose. As we describe the requirements, we build up a vocabulary of *evaluation dimensions*, which categorize the aspects of causes that are important to deciding an explanation's usefulness for a given task. We show at the end of this paper how those dimensions are used to implement task-based evaluation in the program ACCEPTER.

We consider that explanations have the form of a belief-support chain [Schank and Leake, 1989]. Belief-support chains are belief dependency networks, tracing inferences that lead from a set of premises to the outcome being explained. The inferences are plausible connections, not deductions: a belief-support chain increases the tendency to believe the outcome, but does not prove that it must occur.

The sections below sketch eight tasks that drive explanation, and examine the types of information they require from an explanation. We concentrate on tasks that arise when people explain surprising events for their own benefit, going beyond simply trying to make sense of those events.

Learning when to predict the event

When an event is surprising, it may be important to learn how *not* to be surprised by it in the future— how to predict it before it happens next time. For example, a college admissions officer might try to explain why a student who had seemed promising had dropped out, to better predict problems when next looking at applications.

Explanations useful for future predictions must have four properties. First, the links from the explanation's premises to the outcome must be strong enough to make the explainer expect the same outcome, the next time the premises recur. The degree to which an explanation licenses future predictions is its *predictive force*.

Second, the premises must be factors that the predictor is likely to know about in the future. For example, suppose a gambler explains a team's surprising loss by their lack of

concentration. Since he is unlikely to know the team's concentration level in advance of future games, the explanation probably will not help him predict future losses in time to profit. However, a coach might be able to tell in advance, from observing the team in the locker room, so he might be able to use the explanation to predict before future games. How easy it is for an actor to recognize that a premise holds is its *knowability*. But the usefulness of knowable causes depends on a both knowability, and a third property, their *timeliness*: they need to happen far enough in advance for the explainer to benefit from revising his predictions. For example, it would not help the gambler to know that the team would lose after he saw the first play, if he had to bet before the game started.

Finally, the explanation can only be used predictively if it shows an unusual factor of the situation. Even if a team's lack of concentration is a contributing factor to its loss, accounting for the loss by bad concentration will not be helpful if the team never concentrates— what the explanation needs to focus on is the unusual aspect of the situation (*e.g.*, that their superstar was injured). Thus an explanation must trace a surprising event to at least one cause with *distinctiveness*.

Controlling future occurrence of the surprising event

Obviously, preventing future occurrence of an event involves finding premises with *causal force*— that cause the outcome— and that the explainer can block (*controlability*). However, this is not sufficient. For example, if someone burns a cake he is baking, he knows the basic cause of the burning— that the cake became too hot— and how to prevent it— turning down the oven. However, he still needs to know *when* to turn down the oven in the future. If he uses a lower temperature for everything, the things that used to come out perfectly will be underdone. Thus for an explanation to help in preventing an outcome, it must show when to apply the preventative steps: it must allow prediction of the bad outcome, early enough to take steps to block it.

To learn how to achieve the outcome that was surprising, an explainer needs to find a set of causes that are all either controllable, or that have *routineness*, so that they are likely to hold in the future, even without the actor taking action to achieve them himself.

Assigning responsibility and blame

If an actor could have prevented an event's occurrence, or contributed voluntarily to its causes, he bears some responsibility. Depending on the *desirability* of the outcome, the actor may be blameable. He can also be blamed, even if he could not predict or control an outcome, if he contributed to it through an undesirable act. For example, we might blame a drug dealer for an addict's death by overdose, even if overdose deaths are relatively unlikely.

Focusing repair of an undesirable current situation

In order to fix a problem, we need to find problems that both have causal force, and *repairability*. We could explain any automobile breakdown by "there's something wrong with the engine," but the only repair possible at that level of detail is to replace the entire engine, which is not within the financial constraints of most drivers. In addition, we need to find a cause with *independence* from prior causes: if a burned-out transformer in a television is caused by a short circuit, replacing only the transformer will not be an effective repair: unless the short circuit is also fixed, a new transformer would immediately burn out also.

Focusing repair is an instance of a more general category of explanation task: clarifying the current situation to choose an appropriate response. The explainer needs to find an explanation that allows selection of a feasible plan, or the choice between competing alternatives. The competing plans determine which distinctions an explanation must make. For example, an insurance company might try to determine whether a death was suicide or not, in order to determine whether to pay the claim, or refuse. If an explanation for the death shows that the victim were killed in the crash of an airliner, the company would not need further information. However, a lawyer for the family might still want to explain the crash further, to determine whether negligence was involved, and sue those who were responsible.

Sketch of additional purposes

We briefly sketch some additional purposes for explanation, that also affect the type of information a good explanation must provide:

- **Learning a new plan, or refining plan selection**: We can sometimes learn new plans by explaining surprising actions. For example, if we ride home with someone who takes an unusual route, we might explain that the route avoids rush hour traffic, and start using it ourselves. This task for explanation is investigated in [Mooney and DeJong, 1985], which argues that explanations for learning new plans must account for actions in terms of known plan schemas.

- **Changing others' view of an outcome**: An explainer might try to focus on causes that absolve an actor of blame (which is closely related to the task studied in excuse theory), or causes that associate an effect with something desirable or undesirable, to make people see the causes in a new light. For example, if someone took a wrong turn, he might try to make his passengers take a more positive view of the incident, by attributing the turn to distraction because conversation in the car was so captivating.

- **Testing or extending a special-purpose theory**: In order to test a theory, we might require that an explanation use a particular class of rules. For example, an economist might explain layoffs to substantiate his economic theory, by showing it would have predicted them.

ACCEPTER

ACCEPTER is a story understanding program that requests explanations when it encounters anomalies, and judges user-selected explanations both in terms of whether they resolve the anomaly, and in terms of whether they provide the information needed for user-selected goals (see [Leake, 1988] for an overview of the system). Thus ACCEPTER makes the judgement needed for an EBL system to assure that it starts from an explanation relevant to its goals. ACCEPTER was developed as part of SWALE (*e.g.*, [Kass *et al.*, 1986]), a system that uses ACCEPTER's judgements to determine which explanations to accept and generalize for future use.

ACCEPTER implements simple heuristics for judging explanations' premises along the dimensions identified above: predictive force, knowability, timeliness, distinctiveness/routineness, desirability, repairability, and independence. These heuristics allow it to evaluate explanations for four of the purposes above: learning to predict the outcome in the future, repairing device defects, preventing recurrence of the outcome, and assigning blame.

The example below shows ACCEPTER's evaluation of two plausible explanations for a hypothetical Audi recall:

1. The mechanical problems resulted from the car being manufactured by Acme Car Company, under contract as a supplier, due to Acme's bad quality control.

2. The defect resulted from a flaw in the transmissions, which aren't checked by Audi's quality control department.

It would be possible for both the explanations to reflect the facts of the recall. However, the following output, in which ACCEPTER evaluates their usefulness for repairing the defect, shows that they are not equally useful to a mechanic. A mechanic needs to find a state that he can repair— causes that happened in the past, and no longer affect the situation, are unimportant to him, because past events can no longer be repaired. Although the first explanation gives information on factors that led to the defect, it doesn't show a continuing cause that can be repaired in the current situation, so that explanation is useless for him:

```
Checking detail for repair.
  To aid in repair, explanation must show a cause that:
     1. Is repairable.
     2. Is predictive of the problem occurring.
     3. Will not be restored by another state if repaired.

      Checking whether some antecedent satisfies the following tests:
      CAUSAL FORCE TEST (does fact cause consequent?),
      REPAIRABILITY, PREDICTIVENESS, and INDEPENDENT CAUSE.

         Applying test for REPAIRABILITY to AUDI'S PRODUCTION-CONTRACT to ACME.

            Searching up abstraction net for pointers to standard repair plans.
            ... test failed.

         Applying test for REPAIRABILITY to ACME'S NOT M-QUALITY-CONTROL.

            Searching up abstraction net for pointers to standard repair plans.
            ... test failed.

         ... Detail is unacceptable.
```

The second explanation involves two factors that continue to contribute to the car's bad condition: the transmission's defect, and the fact that it is part of the car. ACCEPTER finds that a plan exists for correcting one of them, so that a repair can be done:

```
Applying test for REPAIRABILITY to TRANSMISSION-743'S
PART-OF-RELATIONSHIP to AUDI'S ENGINE.

   Checking repairability of features of TRANSMISSION-743'S
   PART-OF-RELATIONSHIP to AUDI'S ENGINE.
```

```
Searching up abstraction net for pointers to standard repair plans.

AUDI'S ENGINE AS CONTAINER OF TRANSMISSION-743'S
PART-OF-RELATIONSHIP to AUDI'S ENGINE is repairable, since
CONTAINERs of PART-OF-RELATIONSHIPs can usually be repaired
by the standard plan REPLACE-COMPONENT.
... test passed.

... Detail is acceptable.
```

Thus even if both explanations are accurate, an explanation-based system doing repair needs to reject one, and use the other.

Conclusion

Judging explanations according to explainer task provides a way of deciding when to accept one of the many explanations that can be constructed for an event, and finding out what information an incomplete explanation lacks. For example, after a robbery, different tasks make different information important to find out: the victim might focus on what made him a target, to prevent it next time; a policeman might focus on the lack of patrols enabling the crime, to blame his superiors; a social worker rehabilitating the robber might focus on the robber's motivations, to decide how to proceed. The range of purposes and explanations possible for a single event has been discussed in philosophical works such as [Hanson, 1961], but has not been addressed in research on EBL or attribution.[3]

In addition to helping to choose between complete explanations, task-based criteria allow an explainer to use partial explanations in a principled way. Even if the explanation does not completely account for an outcome, it can be useful. For example, if we know just one of the factors that contributed to an event, we may be able to block its occurrence: people who know that high-fat diets contribute to heart attacks can lower their risk, even if they do not know all the other factors necessary to predict heart-attacks. However, traditional approaches to EBL require that learning start from a complete explanation, giving necessary and sufficient conditions for an event, which will be impossible to generate in many situations.

Goal-based focusing is needed because of the complexity of real-world situations. No real-world explanation can include *all* the causally-relevant factors in an event, so that explanations necessarily highlight a few causes out of many. If those causes are irrelevant to the explainer's goals, the explanation will be useless. In order for explanation-based processing to be effective in complex situations, AI systems need to be able to identify which of the many causes of an event are important to their goals, and require that those causes be highlighted in the explanations they use.

[3][Souther *et al.*, 1989] presents an argument close in spirit to ours— that it is essential to be able to generate explanations from a given viewpoint— and identifies classes of explanations that students might seek when studying college-level botony. However, since they discuss only tutoring applications, they do not connect their classes to over-arching goals that make them important, beyond simply doing well in a course.

References

[DeJong, 1988] G. DeJong. An introduction to explanation-based learning. In H.E. Shrobe, editor, *Exploring Artificial Intelligence: Survey Talks from the National Conferences on Artificial Intelligence*. Morgan Kaufmann, Palo Alto, 1988.

[Hammond, 1986] K.J. Hammond. *Case-based Planning: An Integrated Theory of Planning, Learning and Memory*. PhD thesis, Yale University, 1986. Technical Report 488.

[Hanson, 1961] N. Hanson. *Patterns of Discovery*. Cambridge University Press, Cambridge, 1961.

[Heider, 1958] F. Heider. *The Psychology of Interpersonal Relations*, volume XV of *Current Theory and Research in Motivation*. John Wiley and Sons, New York, 1958.

[Kass *et al.*, 1986] A. M. Kass, D. B. Leake, and C. C. Owens. Swale: A program that explains. In *Explanation Patterns: Understanding Mechanically and Creatively*, pages 232–254. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[Kedar-Cabelli, 1987] S.T. Kedar-Cabelli. Formulating concepts according to purpose. In *Proceedings of the Sixth Annual National Conference on Artificial Intelligence*, pages 477–481, Seattle, WA, July 1987. AAAI.

[Keller, 1988] R. Keller. Operationality and generality in explanation-based learning: Separate dimensions or opposite endpoints? In *Proceedings of the 1988 AAAI Spring Symposium on Explanation-based Learning*. AAAI, 1988.

[Kelley, 1967] H. H. Kelley. Attribution theory in social psychology. In D. Levine, editor, *Nebraska Symposium on Motivation*, pages 192–238. University of Nebraska Press, Lincoln, 1967.

[Leake, 1988] D.B. Leake. Evaluating explanations. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 251–255, Minneapolis, MN, August 1988. AAAI, Morgan Kaufman Publishers, Inc.

[Mehlman and Snyder, 1983] R. Mehlman and C. Snyder. Excuse theory: A test of the self-protective role of attributions. *Journal of Personality and Social Psychology*, 49(4):994–1001, 1983.

[Mitchell *et al.*, 1986] T.M. Mitchell, R.M. Keller, and S.T. Kedar-Cabelli. Explanation-based generalization: A unifying view. *Machine Learning*, 1(1):47–80, 1986.

[Mooney and DeJong, 1985] R. Mooney and G. DeJong. Learning schemata for natural language processing. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 681–687, Los Angeles, CA, August 1985. IJCAI.

[Schank and Leake, 1989] R.C. Schank and D.B. Leake. Creativity and learning in a case-based explainer. *Artificial Intelligence*, (40), 1989.

[Souther *et al.*, 1989] A. Souther, L. Acker, J. Lester, and B. Porter. Using view types to generate explanations in intelligent tutoring systems. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, pages 123–130, Ann Arbor, MI, August 1989. Cognitive Science Society.

Are there developmental milestones in scientific reasoning?[1]

Anne L. Fay — Carnegie Mellon University
David Klahr — Carnegie Mellon University
Kevin Dunbar — McGill University

Abstract

This paper presents a conceptual framework that integrates studies on scientific reasoning that have been conducted with different age subjects and across different experimental tasks. Traditionally, different aspects of scientific reasoning have been emphasized in studies with different aged subjects, and the different literatures are somewhat unconnected. However, this separation leads to a disjointed view of the development of scientific reasoning, and it leaves unexplained certain adult behaviors in very difficult scientific reasoning contexts. In this paper we attempt to integrate these three approaches into a single framework that describes the process of scientific reasoning as a search in an hypothesis space and an experiment space. We will present the results from a variety of studies conducted with preschool, elementary school, and adult subjects, and will show how differences in performance can be viewed as differences in the knowledge and strategies used to search the two spaces. Finally, we will present evidence showing that, in sufficiently challenging situations, adults exhibit deficits of the same sort that young children exhibit, even though one might have expected that these developmental milestones were long since passed.

Experimental studies of the development of scientific reasoning skills have produced three distinct and somewhat disjoint literatures. Studies focusing on what Klayman and Ha (1987) call "positive test bias" (the tendency to seek instances that are expected to confirm one's current hypothesis) have concentrated on adult performance; studies on subjects' faulty strategies for the "coordination of theory and evidence" (Kuhn, 1989) have been conducted primarily with adolescents; and studies examining the understanding of necessity and possibility have been conducted with preschoolers. Rarely is one of these phenomena studied in a different age group (i.e., we know of no studies focusing on adults' understanding of the logic of indeterminacy, nor any of preschoolers' positive test bias.)

One possible justification for the different foci is that there might be a sequence of developmental milestones in the acquisition of a complete set of scientific reasoning skills. If so, then it would be prudent for investigators interested in different age levels to address the most obvious inadequacies of their subjects. However, this separation leads to a disjointed view of the development of scientific reasoning, and it leaves unexplained certain adult behaviors in very difficult scientific reasoning contexts. In this paper we attempt to integrate these three approaches into a single framework that describes the process of scientific reasoning as a search in an hypothesis space and an experiment space. We will present the results from a variety of studies conducted with preschool, elementary school, and adult subjects, and will show how differences in performance can be viewed as differences in the knowledge and strategies used to search the two spaces. Finally, we will present evidence showing that, in sufficiently challenging situations, adults exhibit deficits of the same sort that young children exhibit, even though one might have expected that these developmental milestones were long since passed.

Components of Scientific Reasoning

Klahr & Dunbar (1988) have conceptualized the process of scientific reasoning as a dual search in an experiment space and an hypothesis space. Figure 1 depicts the two spaces and the logical relations between them. The upper box is the hypothesis space, which consists of specific hypotheses related to the domain. The lower box is the experiment space. Within this space are the experiments that can be conducted in the domain. The arrows connecting the boxes in the two spaces specify how the experimental outcomes bear on the hypotheses. The heavy arrow between Hypothesis A and Experiment 1 indicates that only Hypothesis A is consistent with the outcome of Experiment 1. This reflects a

[1] Address correspondence to Anne L. Fay Department of Psychology, Carnegie-Mellon University, Pittsburgh, PA 15213, USA. E-mail address: fay@psy.cmu.edu. The first author was supported by a Post-doctoral Fellowship from the James S. McDonnell Foundation Program in Cognitive Studies for Educational Practice. The second author was supported in part by the Personnel and Training Research Program, Psychological Sciences Division, Office of Naval Research, Contract N00014-86K-0349, and in part by grants from NICHHD (R01-HD25211-01A1) and the A.W. Mellon Foundation. The third author was supported in part by a grant from the Natural Sciences and Engineering Research Council of Canada, grant number OGP0037356

Determinate relation. The light arrows between Hypothesis A and Experiment 2 and between Hypothesis B and Experiment 2 indicate that both Hypothesis A and Hypothesis B are consistent with the outcome of Experiment 2. This reflects an *Indeterminate* relation, whereby the outcome of Experiment 2 cannot discriminate between Hypothesis A and Hypothesis B. The absence of arrows between Hypothesis B and Experiment 1, Hypothesis B and Experiment 3, and Hypothesis A and Experiment 3, indicates that these hypotheses are inconsistent with each of these outcomes, reflecting an *Impossible* relation.

The goal of scientific discovery is to generate experiments and hypotheses that will eventually result in a relation of determinacy, whereby only one hypothesis remains consistent with all the experimental outcomes.[2] Thus, the components of scientific reasoning consist of: 1) Identification and understanding of the relations between experiments and hypotheses (i.e., understanding the logic of necessity and possibility); 2) Generation of informative experiments,(i.e, generating experiments that further specify the relations between the two spaces so as to prune the search in the space of hypotheses); 3) Hypothesis generation and revision (i.e., generation of hypotheses from either analogy or via induction from experimental outcomes.)

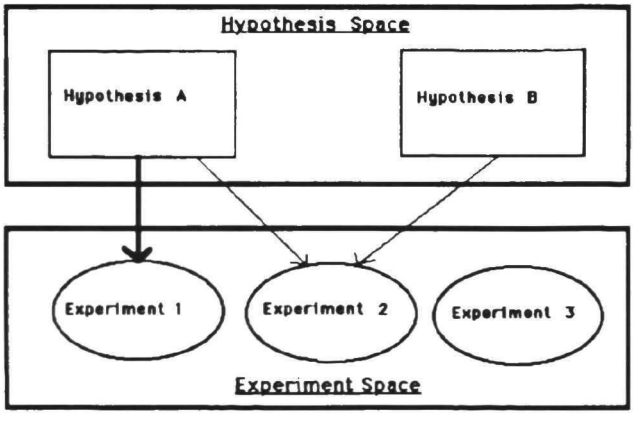


Figure 1. Schematic representation of the two spaces and the relations between them. Heavy arrow indicates a determinant relation between experimental outcome and hypothesis. Light arrows indicate an indeterminate relation and no line indicates an impossible relations.

These three general abilities are fundamental to the process of scientific reasoning, and their deficits are characteristically associated with specific age groups. Adults recognize and understand the implications of indeterminacy, and have heuristics for designing informative experiments, but are notoriously biased toward confirmation in rule discovery tasks (Gorman & Gorman, 1984; Gorman, Stafford & Gorman, 1987; Wason, 1960, 1968).[3] Adolescents, like adults, understand the notion of indeterminacy, but in addition to their bias toward confirmation, they lack the strategies and knowledge for designing informative experiments. Preschool children demonstrate all these deficits, but also show a failure to recognize and/or understand the the implications of determinate vs indeterminate situations. Thus, acquisition of these three abilities might be viewed as *milestones* in the development of scientific reasoning skills. However, the picture is not that straightforward, as we shall argue below. First, however, we will further elaborate each of the three components enumerated above.

Identifying the relation between experiments and hypotheses

One of the basic components of scientific reasoning is the ability to recognize and understand the implications of confirming, and disconfirming evidence. Understanding the implications of these conditions is based on the distinction between determinate and indeterminate outcomes. In Indeterminate situations, the evidence is insufficient to discriminate one hypothesis from another. Until this concept is available, the process of scientific discovery will be severely flawed. Failing to recognize a situation as indeterminate will result in premature termination of the generate-experiment process because a confirming instance will be erroneously identified as sufficient to accept a theory.

Research with preschool children has shown they lack the concept of indeterminacy. In terms of Figure 1, they fail to realize the relation between Experiment 2 and the Hypothesis space. In a study extending Pieraut-Le Bonnic's (1980) investigations of childrens' understanding of possibility and necessity, Fay & Klahr (1990), presented kindergarten children with two boxes of building materials, and a series of objects, one at a time, made from materials taken entirely from one box or the other. For example, Box A might contain sticks and curves and Box B might have sticks and squares. A probe object comprised of

[2] In real-world situations, one never has a determinate relation, as there are an infinite number of possible hypotheses, and there is always te possibility that new evidence will disconfirm the current hypothesis. Nonetheless, the goal of science can be seen as the elimination of all *current* competing hypotheses until only one remains consistent with the existing evidence.

[3] But see Farris & Revlin (1989) for a novel reinterpretation.

only sticks would be *indeterminate* because it could have been constructed from either box. A *determinate* probe object would be one constructed from sticks and curves (only Box A could have been used to make it). The children were asked whether they could tell which box was used to make the object .

Figure 2 shows a schematic of the problem. As can be seen, the task can be mapped directly onto the components shown in Figure 1, with the boxes representing hypotheses and the probe objects representing experiments. Thus in this context the child is presented with a finite hypothesis space (e.g. box with sticks and curves vs. box with sticks and squares) and an experimental outcome (stick & curve object vs. stick-only object) but must determine the relations that exist between them (stick & curve object is determinate vs. sticks-only is indeterminate). *All* the children correctly identified the determinate situation, but only 53% consistently identified the indeterminate situation. Those who failed to recognize the indeterminate situation misidentified it as determinate, claiming that they *could* tell for sure which box had been used to construct the indeterminate probe object.

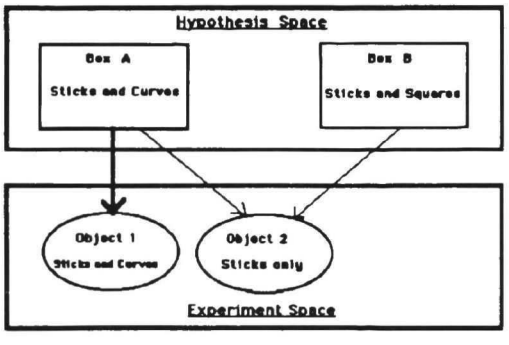


Figure 2. Experimental setup for the Possibility-Necessity study. Heavy arrows indicate a determinate relation and lights arrows indicate an indeterminate relations.

This failure can be explained, in part, by a lack of understanding of the concept of logical necessity. Evidence for this interpretation comes from children's justifications for their responses on determinate problems, which were coded as being based on either positive or negative reasoning. Children were scored as using positive reasoning if their justification was based on the confirmatory relation between the determinate box and the object (e.g. "you used this box because it has sticks"). They were scored as using negative reasoning if their justification referred to the necessity of disconfirmation of the other response (e.g."you had to use this box because the other box doesn't have any curves"). Negative reasoning implies an understanding of logical necessity. That is, it suggests that the child recognizes the insufficiency of confirmation alone and therefore searches the entire hypothesis space to determine the other experiment-hypothesis relationships. Table 1 shows the relation between the type of reasoning used on the determinate problems and performance on the indeterminate problems. The results suggest that the tendency to use negative reasoning is related to the recognition of indeterminate situations.

TABLE 1: Children who use negative reasoning on determinate problems are more likely to be correct on indeterminate problems.

	PERFORMANCE ON INDETERMINATE PROBLEMS (cell entries are number of responses)	
REASONING ON DETERMINATE PROBLEMS	Incorrect	Correct
Positive Reasoning	22	20
Negative Reasoning	6	19

(Chi Square=5.19, p<.025)

Young children demonstrate a lack of understanding of logical necessity, a prerequisite of scientific reasoning. Failing to recognize an indeterminate situation, or to understand its implication, will result in a premature termination of the search based on finding a confirmatory relation between data and theory.

Generating Informative Experiments

The ability to recognize the relations between hypotheses and experiments can be seen as a prerequisite for the skill of generating informative experiments. Informative experiments are designed for the purpose of pruning the hypothesis tree, that is, eliminating impossible hypotheses, and reducing the set of consistent hypotheses. In this situation, the subject is provided with an hypothesis, or enters with a prior hypothesis, and must generate experiments which will lead to the confirmation or disconfirmation of the hypothesis. In reference to Figure 1, the subject is provided with an hypothesis (e.g. Hypothesis B), and the task is to generate experiments that will either disconfirm the hypothesis (e.g. E3), or will discriminate between existing confirming hypotheses (e.g. Experiment 1). Subjects want to avoid writing experiments that fail to discriminate between existing hypotheses (e.g. Experiment 2), or at the least, recognize them as being undiscriminating. Thus, this ability is dependent upon the ability to recognize and understand the relations between

experiments and hypotheses. In addition, it involves an understanding of the goal of experiment generation (reducing the hypothesis tree), and the skills for constructing experiments that will serve these goals.

In a series of experiments with children (8 to 13 years old), and adults, we examined subjects' ability to generate informative experiments (Klahr, Dunbar & Fay, 1990; Klahr, Fay & Dunbar 1990). Subjects were trained to operate a simple programmable device by entering commands (for moving forward, backward, turning right and left, and firing its cannon) and then pressing a GO key to execute the program. This would move an icon on a workstation screen according to the program the subject had entered.[4] Once trained to criterion, they were then asked to discover how an additional function, the REPEAT key, worked. They were then provided with an hypothesis (which was always incorrect), and were asked to write programs to find out if the hypothesis was correct or, if it wasn't, to find out how REPEAT worked.

The design of the study (See Table 2) crossed the plausibility of the given and actual hypotheses. Thus, subjects could be given either a highly plausible or highly implausible hypotheses for how REPEAT worked, and the device was actually programmed to interpret REPEAT in some different, but either plausible or implausible, way. Subjects were given one of the rules and the device actually worked according to a different rule. This effect of these given-actual hypotheses conditions will be expanded on in the following section.

Table 2: Design of "negative feedback" study

		Actual
	Plausible	Implausible
Given		
Plausible	Theory refinement	Theory replacement
Implausible	Theory replacement	Theory refinement

Overall, children performed poorly in discovering the correct rule. Only one-third of the younger children, and half of the older children discovered the rule, compared to 83% of the adults.

One contributing factor to this trend is the degree of informativeness of the programs that the subjects wrote. First, children appear to differ from adults in terms of their awareness of the goals of experimentation. Whereas 83% of the adults made statements referring to experimental design goals, only 20% of the younger children and 47% of the older children made such comments. The quality of these statements also differed. The adults stated experiment goals in terms of increasing the observability and informativeness. The youngest children, on the other hand, primarily made output goal statements (e.g. move it in a square) and some observability goals (e.g. use N=1 otherwise it's too confusing). The older children focused on observability goals (e.g. shorten programs, use easily traced commands).

The above data is based on verbal reports, and as such the tendency to verbalize may be different for the different age groups. A second analysis examined the types of programs that were written. The experimental space for this problem can be viewed along two dimensions, one dimension being the number of commands in the program (lambda) and the other being the magnitude of the argument for REPEAT (N). The ideal experiment is one which maximizes the informativeness of the outcome while minimizing the complexity (i.e. maximizing observability or interpretability of outcome). In the current setting, this means writing minimum length programs that can discriminate the effect of the REPEAT function. By this criterion, the "best" program has a length (lambda) of 3 and a REPEAT argument value (N) of 2. The three age groups differed in their tendency to write programs with these properties. Compared to a random model, the children were 1.5 times *less* likely to generate the ideal experiment whereas the adults were 5 times *more* likely to run such an experiment. In addition, adults were much more systematic in the way that they moved in the experiment space. Their experiments had more of the flavor of a careful experimental series than did the children's.

In summary, children appear to lack the knowledge required to generate informative experiments. Part of this deficit involves a failure to understand the goals of experimentation. Whereas adults' goals were directed toward informativeness and observability, children's goals were directed toward producing a desired effect, and, for the older children, observability. However, even

[4] This "microworld" was a simulated version of the BigTrak, a programmable robot toy that moved around on the ground, originally used by Shrager & Klahr, 1986.

though the older children recognized the importance of observability, they were not overly successful at designing interpretable experiments.

Generating and revising hypotheses

The final component of scientific reasoning is the ability to generate and revise hypotheses in response to experimental outcomes. Combined with the other abilities, this situation can be depicted in Figure 1 by having no boxes specified or present in either the experimental or hypothesis space. Thus all the components of the task must be generated by the subject. In series of studies using the physical BigTrak device, adults and third to sixth grade students were trained on all the functions of the device except the REPEAT and were then asked to write programs to figure out how REPEAT worked (Klahr & Dunbar, 1988; Dunbar & Klahr, 1988). There are two main differences in these studies as compared to the studies mentioned in the previous section: First, in these studies subjects were not given any hypothesis, and had to generate their own hypotheses from the start, and second, there was only one rule for REPEAT, it caused the device to repeat the last N instructions once, where N refers to the argument for REPEAT. There was a strong age effect: 19 of the 20 adults, but only 2 of the 22 children successfully discovered the correct rule, although over half of the children *believed* that they had correctly identified it. Part of the failure can be attributed to the non-informative programs that the children wrote. However, both the adults and the children had the same proportion of experiments from the most informative region of the experiment space, where lambda > N and N > 1. Based on the outcomes from experiments conducted in this region adults were able to induce the correct hypothesis but the children were not.

The hypotheses that were generated by the subjects in this study can be classified into two frames based on the function of N. One frame, Counter-Frame, assigned a role to N where the number indicated how many times something got repeated. The other frame, Selector-Frame, assigned a role to N where the number indicated which instructions got repeated. Children and adults demonstrated an initial preference for the Counter-Frame hypotheses, (which is incorrect in this study). But, whereas adults were able to abandon this hypothesis frame in light of disconfirming evidence and generate the Selector-Frame, the children tended to maintain Counter-Frame hypotheses in spite of disconfirming evidence. Thus the children's search of the hypothesis space was constrained to Counter hypotheses.

Further evidence for this comes from the negative feedback studies described earlier, in which subjects were given an initial hypothesis that could be either from the same frame as the actual hypothesis or from the other frame as the actual hypothesis . Figure 3 shows the effect for Given-Actual Hypothesis conditions. The children were successful when the device worked as a (plausible) Counter, but failed to get the correct rule when it worked as a (implausible) Selector.

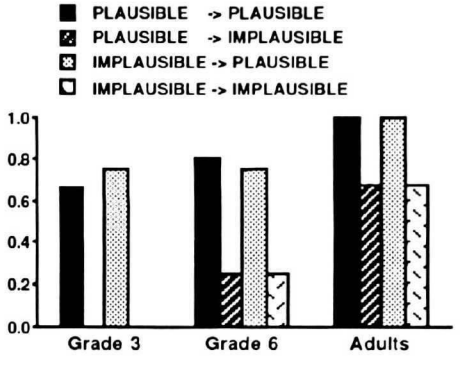


Figure 3. Proportion of subjects discovering correct rule when given a Plausible (counter) or implausible (selector) hypothesis and actual rule was Plausible (counter) or Implausible (Selector).

Prior to running any programs, children and adults differed in terms of there willingness to entertain a Selector hypothesis. Table 3 shows the proportion of subjects in each age group that initially accepted the Given hypothesis or one from the same frame as the Given. The children, especially the younger ones, find the Selector hypothesis very implausible. More than half of the children who rejected the Selector proposed a Counter hypothesis instead. Adults, on the other hand, demonstrated some skepticism over their Given Selector hypothesis, but rather than reject it, they proposed other hypotheses *in addition to* it, and these alternative hypotheses were most likely to be Counters.

Table 3. Proportion of subjects accepting Given Frame prior to running first experiment.

	Given Frame	
Group	Counter	Selector
3rd Grade	1.00	.12
6th Grade	.89	.63
Adults	1.00	1.00

Children's strategies and goals for searching the hypothesis space appear to be different from adults. Children's prior hypotheses constrain their search of the hypothesis space to those areas they consider plausible, in this case, counter hypotheses. Although adults also have prior hypotheses, their search of the hypothesis space is not so constrained, and they will entertain the possibility of implausible hypotheses. Thus adults will abandon a more plausible hypothesis frame given disconfirming evidence, and search the space for a new, less plausible frame, children continue to search within the plausible frame for particular hypotheses that will explain the experimental outcomes.

Milestones or fragile acquisitions?

Given the characteristic deficits associated with each age range, and given the logical necessity for each of the three skills to be in place before the next one can be reliably assessed, it is tempting to view this as a sequence of developmental milestones, in which a skill, once acquired, can be reliably invoked in a wide range of situations and can provide the basis for the subsequent acquisition. However, in other careful analysis of children's strategy acquisition (e.g, Siegler and Jenkins, 1989), it has been shown that the story is not so simple. A new strategy or skill may appear for a while, and then disappear for a protracted period. Or a strategy that seemed quite robust, may, in contexts of sufficient complexity, be abandoned, as subjects revert to simpler, and inadequate, strategies. In the domain of scientific reasoning, we have found just this situation. Dunbar (1989) found that strong prior beliefs about hypotheses can overly constrain search of the hypothesis space, and produce behavior that , at its core, reveals a severely limited ability to discriminate determinate from indeterminate outcomes.

Adult subjects were given training in a simulated molecular genetics laboratory and were shown how to go about discovering how certain genes control the enzyme production of other genes by switching them on when a nutrient is present. This mechanism was *activation.* Subjects were shown the different variables that could be manipulated (e.g. amount of nutrient present, genetic mutations), and how they could use this informatin to run experiments and induce the control mechanism. Subjects were then given a new set of genes and were asked to discover how the enzyme producing genes were controlled. However, the mechanism in this set was *inhibition*: controller genes turn other genes off until a nutrient is present. This can be compared to the Given-Plausible, Actual-Implausible hypothesis condition in the simulated BigTrak studies, as shown in the top-right cell of Table 2. Only 25% of the subjects discovered the inhibition mechanism, similar to the success rate of 6th grade students in the BigTrak study. Subjects often conducted experiments that could have been consistent with many hypotheses, but interpreted the results as confirming their prior (plausible) hypothesis. Sixty-five percent of the subjects remained within the Activation-frame of the hypothesis space, despite experimental evidence that disconfirmed this frame. Thus, like the children in the previous study, their prior hypothesis overly constrained their search of the hypothesis space and also affected their search of the experiment space.

Conclusion

The child-as-scientist view suggests that children go about the world gathering information and building theories (Brewer & Samarapungavan, in press; Karmiloff-Smith, 1988). Other researchers argue that although children may generate theories of their worlds, the process of theory generation and revision is different from that of adults (Kuhn, 1989). The view presented here is that children of different ages have certain characteristic conceptual deficits, which limit their ability to engage in the process of scientific reasoning. We have attempted to show how the three relatively diverse literatures on scientific reasoning can be integrated into a single framework that views discovery as a dual search in a space of hypotheses and experiments, but we have cautioned against a simple view of developmental milestones because of the tendency for people to regress to earlier deficits in sufficiently complex situations. Perhaps this tendency to regress accounts for the substantial educational and institutional supports that provide practicing scientists with the means to maximize the rationality and effectiveness of their efforts at scientific discovery.

References

Brewer, W. F. & Samarapungavan, A. (in press). Child theories versus scientific theories: Differences in reasoning or differences in knowledge? In R. R. Hoffman & D. S. Palermo (Eds.), *Cognition and the symbolic processes: Applied and ecological perspectives* (Vol. 3). Hillsdale, NJ: Erlbaum

Dunbar, K. (1989). Scientific reasoning strategies in a simulated molecular genetics environment. *Proceedings of the 11th annual meeting of the Cognitive Science society*, 426-433. Ann Arbor, MI: Lawrence Erlbaum Associates.

Dunbar, K., & Klahr, D. (1989). Developmental differences in scientific discovery strategies. In D. Klahr & K. Kotovsky (Eds.), *Complex information processing: The impact of Herbert A. Simon.* Hillsdale, NJ: Erlbaum.

Farris, H.H., & Revlin, R. (1989). Sensible reasoning in two tasks: Rule discovery and hypothesis evaluation. *Memory & Cognition*, **17**, 221-232.

Fay, A. L. & Klahr, D. (1990). Cognitive precursors to scientific reasoning: the development of the concepts of possibility and necessity. Working Paper, Dept. of Psychology, Carnegie-Mellon University.

Gorman, M.E., & Gorman, M.E. (1984). A comparison of disconfirmatory, confirmatory, and control strategies on Wason's 2-4-6 task. *Quarterly Journal of Experimental Psychology*, **36a**, 629-648.

Gorman, M.E., Stafford, A., & Gorman, M.E. (1987). Disconfirmation and dual hypotheses on a more difficult version of Wason's 2-4-6 task. *Quarterly Journal of Experimental Psychology*, **39a**, 1-28.

Karmiloff-Smith, A. (1988) The child is a theoretician, not an inductivist. *Mind and Language*, **3**, 183-195.

Klahr, D., & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Science*. **12**, 1-48.

Klahr, D., Dunbar, K. & Fay, A. L. (1990) Designing good experiments to test "bad" hypotheses. In J. Shrager & P. Langley (Eds.), *Computational models of discovery and theory formation.* Morgan-Kaufman.

Klahr, D., Fay, A.L., & Dunbar, K. (1990). *Developmental differences in experimental heuristics.* Working paper, Department of Psychology, Carnegie Mellon University..

Kuhn, D. (1989) Children and adults as intuitive scientists. *Psychological Review*, **96**, 674-689.

Klayman, J., & Ha, Y. (1987). Confirmation, disconfirmation and information in hypothesis testing. *Psychological Review*, **94**, 211-228.

Pieraut-Le Bonniec, G. (1980). *The Development of Modal Reasoning.* New York, NY: Academic Press.

Shrager, J., & Klahr, D. (1986). Instructionless learning about a complex device. *Journal of Man-Machine Studies*, **25**, 153-189.

Siegler, R. S. & Jenkins, E. *(1989) How children discover new strategies.* Hillsdale, N. J.

Wason, P. C. (1960). On the failure to eliminate hypotheses in a conceptual task. *Quarterly Journal of Experimental Psychology*, **12**, 129-140.

Wason, P.C. (1968). On the failure to eliminate hypotheses: A second look. In P.C. Wason & P.N. Johnson-Laird (Eds.), *Thinking and Reasoning.* Middlesex: Penguin Books.

Why Fodor and Pylyshyn Were Wrong: The Simplest Refutation

David J. Chalmers
Center for Research on Concepts and Cognition
Indiana University

Abstract

This paper offers both a theoretical and an experimental perspective on the relationship between connectionist and Classical (symbol-processing) models. Firstly, a serious flaw in Fodor and Pylyshyn's argument against connectionism is pointed out: if, in fact, a part of their argument is valid, then it establishes a conclusion quite different from that which they intend, a conclusion which is demonstrably false. The source of this flaw is traced to an underestimation of the differences between localist and distributed representation. It has been claimed that distributed representations cannot support systematic operations, or that if they can, then they will be mere implementations of traditional ideas. This paper presents experimental evidence against this conclusion: distributed representations can be used to support direct structure-sensitive operations, in a manner quite unlike the Classical approach. Finally, it is argued that even if Fodor and Pylyshyn's argument that connectionist models of compositionality must be mere implementations were correct, then this would still not be a serious argument against connectionism as a theory of mind.

Introduction

The trenchant critique by Fodor and Pylyshyn (1988) threw a scare into the field of connectionism, at least for a moment. Two distinguished figures, from the right side of the tracks, were bringing the full force of their experience with the computational approach to cognition to bear on this young, innocent field. It was enough to get anybody worried for a while. But after the initial flurry, connectionists gradually settled down to the view that while Fodor and Pylyshyn had posed a challenge for the field, it was certainly not an unanswerable one. A spate of "refutations" quickly followed. These generally took two forms: argument (e.g. Clark 1989, Smolensky 1987, van Gelder 1990), or counterexample (Elman 1990, Pollack 1990, Smolensky 1990). (One is reminded of Nietzsche's observation: "It is not the least charm of a theory that it is refutable.")

The point of this paper is to offer a few observations on the whole business. The primary purpose is to offer a particularly simple refutation of Fodor and Pylyshyn's argument that I do not believe has been presented elsewhere. Straightforward considerations about the structure of their argument will show that it cannot have succeeded in its intended purpose. Furthermore, simple as these considerations are, they lead into deeper issues about just *why* their argument was wrong, and about the vital properties of connectionist models that were not taken into account. In particular, the role of *distributed representation* will be gone into. The ability of distributed representations to support structure-sensitive operations will be demonstrated with some experimental results. Finally, this will lead into the issue of the possible *implementation* of Classical ideas by connectionist models, and expose the shortsightedness of some of Fodor and Pylyshyn's claims here.

Refutation

Recall the major thrust of Fodor and Pylyshyn's argument: that connectionist models cannot admit of a compositional semantics. Or, more accurately, not unless they are an implementation of a Classical architecture. Manifestations of compositional semantics are certainly ubiquitous in our thought, particularly in our language, through its *compositionality* (the meaning of "the girl loves John" is a function of the meaning of its constituent parts, "the girl", "loves", and "John"),

and its *systematicity* (the ability to think "John loves the girl" is tied to the ability to think "the girl loves John"). So if connectionism cannot handle compositional semantics, then that's a problem for connectionism.

The refutation of F&P's argument can be stated in one sentence, then explained. *If F&P's argument is correct as it is presented, then it implies that no connectionist network can support a compositional semantics; not even a connectionist implementation of a Turing Machine, or of a Language of Thought.* But this is a problem for F&P, as it is well-known that connectionist networks can be used to implement Turing Machines (or at least Turing Machines with arbitrarily large but finite tape), and it is well-known that Turing Machines can be used to support a compositional semantics. Furthermore, the *human brain* is like a connectionist network in many ways, and the human brain certainly supports a compositional semantics. So if F&P's argument really establishes that *no* connectionist network can support a compositional semantics, then it establishes a false conclusion. So, applying the contrapositive of the italicized sentence above, F&P's argument is not correct as it stands.

Of course, Fodor and Pylyshyn do not *want* to imply such a conclusion. Indeed, they take great care to point out that the best future for connectionism will lie in using it as an *implementation* strategy. Connectionist implementations of Classical systems will certainly support a compositional semantics, if not in a particularly interesting way. Well and good; of course they must say such a thing: it may be slightly embarrassing that the brain is made of neurons and not directly out of symbolic structures, but it is a *fact*, and as a fact it must be dealt with. But what they *say* is one thing. Their actual *argument* is a different matter.

The substantive argument in F&P's paper, that connectionist models cannot support a compositional semantics, takes up only a few pages (pp. 15-28). This starts with a simple localist connectionist network (that is, a network with one node representing one concept). F&P show that this network cannot possibly possess a compositional semantics, and argue that this applies equally to networks with distributed semantics (that is, a network with one concept being represented over many nodes). Therefore, the argument concludes, it is impossible for the semantics of a connectionist network to be compositional, whether these semantics are localist or distributed.

There is something very strange about this conclusion. It is plainly false; it is universally recognized that *some* connectionist networks have compositional semantics: namely, connectionist implementations of Classical architectures. So why are these not excluded from the argument? Going through the argument, the reader expects that at any point soon, there will be an *escape clause* — a clause showing why the argument as it stands does not apply to connectionist implementations of Classical architectures. But this clause never appears; nothing close to it, in fact. F&P are left in the improbable position of having "proved" that even connectionist implementations of Classical models have no compositional semantics. Faced with such a situation, we can only conclude that the argument is defective. Supporters of F&P might argue that the flaw simply lies in the lack of an escape clause, which can easily be supplied; but no such escape clause is in evidence, and the onus lies with these people to provide it. In the meantime, we can conclude that the defect lies elsewhere: very likely, in the generalization from localist to distributed semantics. More on this in a moment, after an analogy.

Say a mad scientist comes up to us with a "proof" that the Earth is the only inhabited planet in the universe. She runs through an impressive *a priori* argument, showing why it is impossible that the right kinds of biochemicals could be assembled in the right way, that the requisite organizational complexity could not arise, and so on. She concludes: life could not have arisen on any planet in this universe. But then, of course, it is an obvious fact that life arose on Earth. "That's OK," she answers, "that suits me fine. We knew that already. So what I've established is that life cannot have arisen anywhere but Earth." Now this will strike us as ad hoc, and as

extremely poor logic. Their main argument never mentioned Earth; there was no *escape clause* showing just why the argument doesn't apply to Earth. To modify the conclusions of one's argument by considerations *external* to the argument is to admit that the argument is faulty. ("Mars is inhabited? OK, our argument demonstrates that life cannot have arisen anywhere but Earth or Mars.") If the argument can be fixed so that Earth is excluded from its force, very likely other planets will be excluded also. Analogously: if F&P's argument can possibly be fixed up so that it excludes Classical implementations from the scope of its conclusion, then the same fixes will probably exclude many other connectionist models too.

Refuting Fodor and Pylyshyn in Four Easy Steps

All this has been a long-winded way of making the following simple argument:

(1) In F&P's argument that no connectionist models can have compositional semantics, there is no escape clause excluding certain models (such as Classical implementations) from the force of the conclusion. (By observation.)

(2) If F&P's argument is correct as it stands, then it establishes that *no* connectionist model can have compositional semantics. (From (1).)

(3) But some connectionist models obviously *do* have compositional semantics; namely, connectionist implementations of classical models. (By observation, accepted by all.)

(4) Therefore, F&P's argument is not correct as it stands. (From (2), (3).)

Summing things up: Let C denote the class of all possible connectionist models, together with all possible associated semantics. Let FP denote the subset of C of models whose semantics are not compositional. Let L denote the subset of C consisting of models with localist semantics. Let IMP denote the subset of C consisting of connectionist implementations of Classical models. The conclusion that F&P *want* to establish is that FP = C – IMP.

In their argument, F&P first establish that L ≤ FP. (Here "≤" denotes set inclusion.) Let us grant them this, though some might argue. They then argue that it makes no difference whether the semantics are localist or distributed. Now, clearly the two possibilities of localist and distributed semantics exhaust the set C, so this argument, if correct, establishes that FP = C. But this is plainly false, as IMP < C but it is not the case that IMP < FP.

We may conclude that *all* F&P have established is that L ≤ FP ≤ C – IMP. The step in the argument that generalizes to *all* distributed semantics is plainly defective. Although F&P would like to hold that it generalizes to all distributed semantics *except* those used to implement Classical models, the burden rests with them to show that this is the case. The conclusion established is a much weaker statement than FP = C – IMP. As things stand, it is just as likely that FP = L as that FP = C – IMP, though no doubt the truth lies somewhere in the middle.

Localist and Distributed Representation

So far, we have given a simple logical demonstration that F&P's argument must be flawed. It remains to precisely locate the weak spot in the argument. Fortunately, this is not hard to do. To find this, we must think about just *why* certain models, implementations and possibly others, slip through the argument's net. By now, no doubt, supporters of F&P are lining up in droves, waiting to say: "But of *course* the argument doesn't apply to implementations of Classical models. Implementations are *different* — the representations of Classical symbols in such a network will not exist at the level of the *node*, but at a much higher level. These symbols will be

able to combine compositionally and autonomously." To such a person we might reply "Congratulations! You have just discovered the power of distributed representation."

Many connectionists have noted that the small localist network that F&P used as their chief example was most unrepresentative of the connectionist endeavour of a whole. When one asks what is the deepest philosophical commitment of the connectionist movement, the answer is surely this: the rejection of the atomic symbol as the bearer of meaning. Connectionists feel that atomic tokens simply do not carry enough information with them to be useful in modeling human cognition. Rather, distributed, subdivisible, malleable representations are the cornerstone of the connectionist endeavour. For this reason, localist networks are regarded by many connectionists as not really connectionist at all. These networks employ precisely the traditional notion of atomic symbols, with a new twist added by connecting all of these by associative links. (We might thus call localist connectionism "symbolic AI with soft constraints.")

The use of a localist network by F&P, then, betrays a lack of understanding of the connectionist endeavor. They believe that nothing depends on the localist/distributed distinction; the connectionist, on the other hand, believes that everything depends on it. To F&P, a connectionist distributed representation is just a spread-out version of a single node (this comes out clearly in the footnote to p. 15). To the connectionist, a group of nodes functioning separately has functional properties far beyond those of an isolated unit. Small differences in the activity of a subset of nodes can make subtle or unsubtle differences to later processing, in a way that no single node can manage. A group of nodes carries far more *information* than a single node, and as such to the connectionist is a far more likely candidate for for semantic interpretation. And most importantly, a distributed representation has a great deal of internal structure. (The point that Fodor and Pylyshyn underestimate the power of distribution is by no means original. It was first made by Smolensky (1987).)

Before moving on, we should briefly examine F&P's demonstration of why their argument applies equally to localist and distributed networks. This will be brief, as the relevant material is brief. On the bottom of p. 15, we find

> To simplify the argument, we assume a more 'localist' approach, in which each semantically interpreted node corresponds to a single Connectionist unit; but nothing relevant to this discussion is changed if these nodes actually consist of patterns over a cluster of units.

No argument to be found there. And later (p. 19)

> To claim that a node is neurally distributed is presumably to claim that its states of activation correspond to changes in neural activity — to aggregates of neural 'units' — rather than to activations of single neurons. The important point is that nodes that are distributed in this sense can perfectly well be syntactically and semantically atomic: Complex spatially-distributed implementation in no way implies constituent structure.

No-one will begrudge F&P this passage. As it stands, it is perfectly true. But it would only be interesting as argument if the last two sentences changed so that the "can" became a "must" and the "in no way implies" became "forbids". But it is precisely this that F&P cannot establish. We can conclude that their argument against distributed representation (and this is the extent of it) is weak. F&P go on to argue against connectionist models whose semantics are "distributed over microfeatures". But, as elsewhere, the kinds of semantics they consider bear little resemblance to those found anywhere in connectionism. This is the fundamental flaw in F&P's argument: lack of imagination in considering the possible ways in which distributed representations can carry semantics. It is a different variety of distributed semantics that would be carried by a connectionist implementation of a Turing Machine (and this, then, accounts for the logical flaw

detailed above.) And it is a different variety again of distributed semantics that can yield connectionist models of compositionality in important new ways.

It is no accident that three of the most prominent *counterexamples* to F&P's argument — the models of Elman, Pollack, and Smolensky — all use distributed representation in an essential way. Smolensky's tensor-product architecture simply could not work in a localist framework. Its multidimensional tensor representations are by their nature spread over many nodes. Elman's implicit structure which develops in a recurrent network could also not succeed in a localist framework — the many subtle adjustments needed for various syntactic distinctions to develop could not be made. And Pollack's Recursive Auto-Associative Memory has a deep commitment to distribution — if it were one-concept-to-one-node, then its recursive encoding scheme could never get off the ground.

Structure-Sensitive Operations on Distributed Representations

The Classicist might now reply: "All this talk of distributed representations is all very well. Maybe you can *encode* compositional information into such a representation. But can you *use* it?" This point is initially plausible. If the structural information is present but cannot be processed, then it is useless. The Classicist might hold that connectionist compositional structure might be buried too deeply, too implicitly, to be accessed in a useful way. Indeed, in a recent paper, Fodor and McLaughlin (forthcoming) argue that to support structure-sensitive processing, a compositional representation must be a concatenation of explicit tokens of the original constituent parts. If this argument is correct, then connectionist representations that represent structure only in a distributed, implicit way will not have the causal power to support structure-sensitivity.

One obvious reply that the connectionist might make is that clearly *some* structure-sensitive operations can be supported by such representations: namely, the operation of extraction of the original constituents. Both Smolensky's and Pollack's models, for instance, include decodal processes that go from a compositional representation back to its parts. This reply, while valid, is not very interesting. If structure-sensitive processing must always proceed through an initial stage of decomposition into constituents, then what we are dealing with is essentially a connectionist implementation of a Classical symbol processing. In such processing, distributed representation is used as a mere implementational technique.

Fortunately, this is not always the case. In fact, distributed representations of compositional structure *can* be operated on directly, without proceeding through an extraction stage. This offers the promise of a connectionist approach to compositionality that is in no sense an implementation of the Classical notion. (It should be noted that Pollack and Smolensky have addressed this issue briefly in their models, but in a more limited way than outlined below.)

I have performed a series of experiments demonstrating the possibility of effective structure-sensitive operations on distributed representations. I can only outline them very briefly here; they are presented in more detail in (Chalmers, forthcoming). The experiments used a Recursive Auto-Associative Memory (RAAM; see Pollack 1988, 1990) to encode syntactically structured

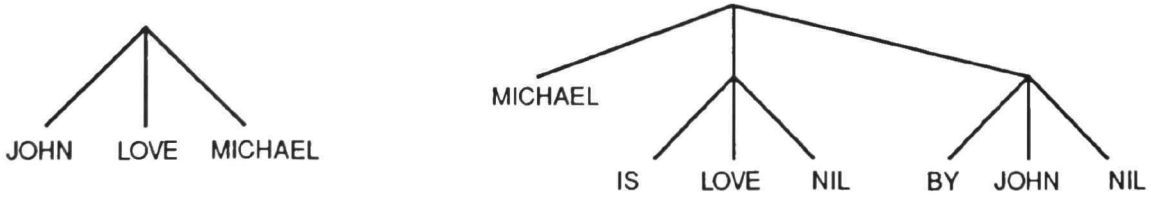


Figure 1. Examples of sentences to be represented.

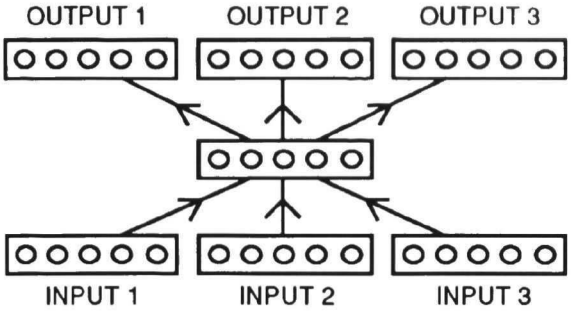


Figure 2. The basis of the RAAM network.

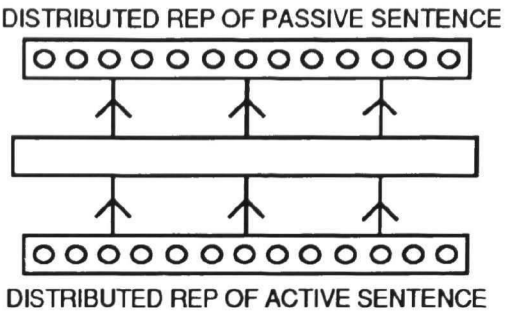


Figure 3. The Transformation network.

representations of sentences in distributed form. Following this, a back-propagation network learned to perform syntactic transformations directly from one encoded representation to another.

The sentences represented were all of similar syntactic form to "John loves Michael" (active) or "Michael is loved by John" (passive). Five different names/verbs were used as fillers for each slot of subject, verb or object, giving 125 possible sentences of each type altogether. These sentences were assigned syntactic structure as shown in Figure 1. A RAAM network was trained to encode 125 sentences of each kind into a distributed form. (Pollack 1990 gives details of the RAAM architecture.) This is done by assigning each word a primitive localist representation (over 13 units), and then training a 39-13-39 backpropagation network (Figure 2) to auto-associate on the three leaves descending from every internal "node" (in the trees in Figure 1).

This gives us a 13-node distributed representation of the three leaves. Where necessary, this 13-node distributed representation is repropagated as part of the input to the 39-13-39 network, leading to higher-order structures being encoded. Eventually, we have a distributed representation of the entire tree. This process can be used, in principle, to encode any tree of valence 3 recursively.

The RAAM network learned to represent all 250 sentences satisfactorily, so that the distributed encodings of each sentence could be decoded back to the original sentence. These distributed representations were then used in modeling the process of syntactic transformation. In particular, the transformation of *passivization* was modeled: that is, the passing from sentences like "John loves Michael" to sentences like "Michael is loved by John". (No commitment to any particular linguistic paradigm is being made here; syntactic transformations are simply being used as a clear example of the kind of structure-sensitive operations with which connectionist models are supposed to have difficulty.)

150 of the encoded distributed representations (75 active and and the corresponding 75 passive sentences) were randomly selected for the training of the Transformation Network. This was a simple 13-13-13 backpropagation network (Figure 4), which took a representation of an active ("John loves Michael") sentence as input, and was trained to produce a representation of the corresponding passivized sentence ("Michael is loved by John") as output.

Training proceeded satisfactorily. The interesting part was the test of generalization, to see if the network was truly sensitive to the syntactic structure encoded in the distributed forms. The Transformation network was tested on the 100 remaining sentences from the original corpus. The 50 active sentences were encoded by the RAAM and fed to the Transformation network, yielding a 13-node output pattern. This was fed to the RAAM network for decoding. In all 50 cases, the output pattern decoded to the correct passivized sentence. Thus, not only was the Transformation network able to be trained to optimal performance, but the generalization rate on

new sentences of the same form was 100%. The reverse transformation was also modeled (from passive to active). Performance was equally good, with a generalization rate of 100%.

These results establish without doubt that it is possible for connectionist networks to model structure-sensitive operations directly upon distributed representations. This bears on the arguments at hand in two ways.

(1) It demonstrates that not only can compositional structure be *encoded* in distributed form, but that the structure implicitly present within the distributed form can be *used* directly for further processing. This provides a direct counterexample to the Fodor and McLaughlin argument. Despite the lack of explicit concatenative structure in the RAAM representations, they support structure-sensitive processing anyway.

(2) It demonstrates the possibility of structure-sensitive operations in connectionist models which are in no sense implementations of Classical algorithms. To see this, note that when a structure-sensitive operations is being performed upon a Classical compositional representation, all processing *must* first proceed through a step of explicit decomposition, with particular tokens being explicitly extracted. In the connectionist model above, the transformation operation takes place without ever having to extract those constituent parts. Instead, the operation is direct and holistic.

The Relationship Between the Approaches

A argument made frequently by Fodor and Pylyshyn is that connectionists have two choices: either (1) ignore the facts of compositionality and systematicity, and thus have a defective theory of mind, or (2) accept compositionality and systematicity, in which case connectionism merely becomes a strategy for implementing Classical models. The following passage is typical:

> ...if you need structure in mental representations anyway to account for the productivity and systematicity of minds, why not postulate mental processes that are structure sensitive to account for the coherence of mental processes? Why not be a Classicist, in short? [p. 67]

This argument is rather curious. It is not only that it contradicts the evidence, demonstrated above, that connectionism might model structure-sensitive processes in a non-Classical way. There is also a deeply-embedded false assumption here: the assumption that *compositionality is all there is*.

To see the role that this assumption plays, shift the temporal position of the debate back a few decades. Let us imagine two traditional behaviorists, Fido and Pavlovian, who are rather distressed at the current turn of events. The revolutionary "cognitivists" have recently appeared on the screen, and are doing their best to undermine the basic assumptions of decades of solid research in psychology. Our behaviorists have difficulty grasping the idea of this movement. They express their bewilderment as follows: "Surely you all recognize that Classical Conditioning is a fact of human nature. The empirical evidence is overwhelming. But your cognitivist ideas do not take it sufficiently into account. There is no guarantee of stimulus-response association in your models as they stand. It seems to us that you have two choices: either (1) ignore the facts of Classical Conditioning, and therefore have a defective theory of mind, or (2) accept Conditioning and stimulus-response association, in which case cognitivism merely becomes a strategy for implementing the Behaviorist agenda."

Presumably a cognitivist (such as Fodor or Pylyshyn) would quickly see the flaw in this argument. To be sure, Conditioning is an empirical fact, and any complete theory must account for it. But it's certainly not the *only* fact, or even the most important fact, about the human mind. The cognitivists may pursue their own research agenda, making progress in many areas, and

paying as much or as little attention to Conditioning as they like. Eventually they will have to come up with some explanation of the phenomenon, and who knows, it may well end up looking much like the Behaviorist story, *as far as conditioning is concerned.* But this doesn't mean that the cognitivist theory of *mind* looks much like the behaviorist theory overall, for the simple reason that *conditioning is only one part of the story*.

Similarly, compositionality is only one part of the story. Connectionists are free to pursue their own agenda, explaining various aspects of the mind as they see fit. Sooner or later, they will have to explain how compositionality fits into the picture. The story that connectionism tells about compositionality may prove quite similar to the Classical picture, or it may prove different. But even if it proves similar, this diminishes the status of connectionism not at all. The fact that connectionism might implement Classical theories of *compositionality* does not imply that connectionism would be implementing Classical theories of *mind.* Compositionality is just one aspect of the mind, after all. (Aspects of cognition for which compositionality seems relatively unimportant include: perception, categorization, motor control, memory, similarity judgments, association, attention, and much more. Even within language processing, compositionality is only part of the story, albeit an important part.)

Behaviorism was very good at explaining conditioning, but it had a problem: it was *only* good at explaining conditioning. Fodor and Pylyshyn's Classicism is good at explaining compositionality and compositional semantics, but it's not necessarily good at explaining much else. Both conditioning and compositionality are only small aspects of the mind; it seems to be an illusion of perspective that led to behaviorists and Classicists putting so much respective emphasis on them.

Fodor and Pylyshyn's arguments establish that compositionality *exists*, but for their arguments above to succeed, they would need to establish a rather stronger claim: that compositionality is *everything*. Such a claim is obviously false, so connectionism can go on happily trying to explain those areas of the mind that it chooses to. If the connectionist story about compositionality ends up looking a little like the Classical story, then well and good — it implies that the Classicists haven't been wasting their time completely all these years, and there may be room for a healthy amount of ecumenicism. In the meantime, preemptive relegation of either approach to a subsidiary role is probably a bad idea.

Acknowledgements: Thanks to Indiana University for support, and to Bob French, Liane Gabora and Doug Hofstadter for comments.

References

Chalmers, D. J. (forthcoming). Syntactic transformations on distributed representations. *Connection Science.*

Clark, A. (1989). *Microcognition.* Cambridge, MA: MIT Press.

Elman, J. L. (1990). Structured representations and connectionist models. In Gerald Altmann (ed.), *Computational and Psycholinguistic Approaches to Speech Processing.* New York: Academic Press.

Fodor, J. A.,& Pylyshyn, Z. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28: 3-71.

Fodor, J.A., & McLaughlin, B. (forthcoming). What is wrong with tensor product connectionism? In T. Horgan and J. Tienson (eds.), *Connectionism and the Philosophy of Mind.* Cambridge, MA: MIT Press.

Pollack, J. B. (1988). Recursive auto-associative memory: Devising compositional distributed representations. In *Proceedings of the Tenth Annual Conference of the Cognitive Science Society.* Montreal, Canada, pp. 33-39.

Pollack, J. B. (1990). Recursive distributed representations. *Artificial Intelligence*, forthcoming.

Smolensky, P. (1987). The constituent structure of connectionist mental states: A reply to Fodor and Pylyshyn. *Southern Journal of Philosophy*, 26: 137-163.

Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, forthcoming.

Van Gelder, T. (1990). Compositionality: A connectionist variation on a Classical theme. *Cognitive Science*, 4.

Phonological Rule Induction: An Architectural Solution

David S. Touretzky[1], Gillette Elvgren III[1], and Deirdre W. Wheeler[2]

[1]School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

[2]Department of Linguistics
University of Pittsburgh
Pittsburgh, PA 15260

Abstract: Acquiring phonological rules is hard, especially when they do not describe generalizations that hold for all surface forms. We believe it can be made easier by adopting a more cognitively natural architecture for phonological processing. We briefly review the structure of M^3P, our connectionist Many Maps Model of Phonology, in which extrinsic rule ordering is virtually eliminated, and "iterative" processes are handled by a parallel clustering mechanism. We then describe a program for inducing phonological rules from examples. Our examples, drawn from Yawelmani, involve several complex rule interactions. The parallel nature of M^3P rule application greatly simplifies the description of these phenomena, and makes a computational model of rule acquisition feasible.

1. Introduction

In previous publications [10-14] we described our work on a connectionist approach to phonology inspired initially by the ideas of John Goldsmith [2] and George Lakoff [6,7]. Our "Many Maps Model of Phonology," M^3P, is an attempt to provide a computational account for both the regularities and peculiarities of human phonological behavior. The parallel rule formalism we developed is constrained by properties of an underlying connectionist sequence manipulation architecture. We suggested that rule acquisition should be easier in this parallel formalism than in the classical generative formalism, because rule ordering constraints are virtually eliminated, and iterative rule application, such as required for vowel harmony, is replaced by a single application of a clustering rule.

In this paper we present some experimental evidence to support our claim that rule acquisition is more tractable in the M^3P formalism. We describe a rule-learning program that examines a set of underlying and surface forms and induces rules to account for the differences. Our program is language-independent, but for reasons of space we will confine our discussion to examples from a single language. We choose the Yawelmani dialect of Yokuts (an American Indian language from California) because it combines several interesting processes in a small number of examples: epenthesis, lowering, shortening, and vowel harmony. Our Yawelmani data come from Kenstowicz & Kisseberth [4], who draw on data from Newman [9] and the analysis of Kuroda [5]. Lakoff offers an alternative to the Kenstowicz & Kisseberth analysis in [7].

Of course, humans are not supplied with underlying forms when they learn their language, so in this respect at least, our program is not a realistic model of human language acquisition. However, we view it as an important first step toward more ambitious, human-like models. We are aware of no other phonological rule learning program, connectionist or otherwise, that can deal with complex rule interactions of the sort found in Yawelmani and many other languages. Phonological rule acquisition is simply a hard problem. Our purpose here is to show that it can be made less hard by adopting a

more cognitively natural architecture. We include a detailed discussion of Yawelmani phonology below to underscore the complexity of the task. Non-linguists can safely skim the following section.

2. The Facts of Yawelmani

In the following discussion we will focus on four rules from Yawelmani, all involving vowels. The (underlying) phonemic and (surface) phonetic vowel systems for Yawelmani are given below. Colons denote long vowels. Note that there are no long high vowels (/i:/ or /u:/) at the surface, and that /e/ and /e:/ do not exist phonemically.

Phonemic (underlying)		Phonetic (surface)	
i/i:	u/u:	i	u
	o/o:	e/e:	o/o:
	a/a:		a/a:

For typographical convenience, following Kenstowicz & Kisseberth, the short and long mid vowels are represented here as 'e' and 'o', although phonetically they are actually [ɛ] and [ɔ]. Given this phonemic inventory, we have evidence for the following distinctive feature analysis of the vowels of Yawelmani:

	i	u	o	a	i:	u:	o:	a:
high	+	+	–	–	+	+	–	–
round	–	+	+	–	–	+	+	–
long	–	–	–	–	+	+	+	+

Suffixes in Yawelmani exhibit alternations between non-round and round vowels. The basic pattern is that a vowel is pronounced round (and back) if immediately preceded by a round vowel of the same height, that is, u-i becomes u-u, and o-a becomes o-o. Note the alternation of -hin/-hun and -al/-ol below:

xat-hin	'eats'	xat-al	'might eat'
bok'-hin	'finds'	bok'-ol	'might find'
xil-hin	'tangles'	xil-al	'might tangle'
dub-hun	'leads by the hand'	dub-al	'might lead by the hand'

Rounding applies iteratively, throughout an entire word, as illustrated in the following examples:

Underlying	Surface	Gloss
/dub-wis-mi/	[dub-wus-mu]	'having led each other by the hand'
/t'ul-sit-hin/	[t'ul-sut-hun]	'burns for'

Returning now to the vowel inventories of Yawelmani, it is worth noting again that there are no underlying segments /e/ and /e:/. The surface segments [e] and [e:] are derived from /i:/ through the application of two rules: Lowering and Shortening. All long vowels lower, and vowels in closed syllables shorten. These rules are shown below; the $ denotes a syllable boundary.

Yawelmani Lowering

$$\begin{matrix} V \\ [+long] \end{matrix} \rightarrow [-high]$$

Yawelmani Shortening

$$V \rightarrow [-long] \,/\, _\, C\ \$$$

Evidence for shortening comes from paradigms like the following, which show an alternation in vowel length (such as o:/o or a:/a) in the root. Notice that the short variants are followed by *two* consonants; their syllable structure is CVC$CV(C), with the first vowel satisfying the environment for shortening.

Nonfuture	Imperative	Dubitative	Future	
dos-hin	dos-k'o	do:s-ol	do:s-en	'report'
lan-hin	lan-k'a	la:n-al	la:n-en	'hear'

We now argue, in agreement with previous researchers, that not only does [e] derive from a high underlying vowel, but some occurrences of [o] do as well. In particular, the underlying form of the root [c'om-] "destroy" should be /c'u:m-/. One source of evidence for this is harmony. If the underlying form were /c'o:m-/, the non-future form would be *[c'om-hin], but it is actually [c'om-hun]. Lowering and shortening apply only after harmony has had its effect. This also explains why we get [c'om-al] (no rounding of the suffix vowel), not *[c'om-ol]. Compare this with the root meaning 'report,' which is underlyingly /do:s-/, and triggers harmony as expected, giving [do:s-ol], not *[do:s-al].

c'o:m-al	'might destroy'	do:s-ol	'might report'
c'om-hun	'destroys'	dos-hin	'reports'

The interaction of harmony, lowering, and shortening is illustrated below:

	/c'u:m-al/	/c'u:m-hin/
harmony	—	u
lowering	o:	o:
shortening	—	o
	[c'o:mal]	[c'o m h u n]

There is another set of apparent counterexamples to harmony, in that a sequence of vowels that agree in height at the surface do not agree in rounding (o-e), and vowels that disagree in height do agree in rounding (u-o):

bok'-en	'will find'	xat-en	'will eat'
dub-on	'will lead by the hand'	giy'-en	'will touch'

Notice that in [dub-on] the surface *non*-high vowel of the suffix appears rounded after a *high* vowel in the root. Here again, the regularity in the system reemerges if we assume that the future suffix is underlyingly long and high, i.e., /-i:n/. It surfaces as [-en] through the combined effects of lowering and shortening, or as [-on] if harmony has applied first.

There is one additional rule involving vowels that interacts with vowel harmony, namely epenthesis. There is a class of roots of the form CVCC- which have alternants of the form CVCiC- or CVCuC- depending on the preceding vowel and the effects of harmony.

?ugn-al	'might drink'	?ugun-hun	'drinks'
logw-ol	'might pulverize'	logiw-hin	'pulverizes'

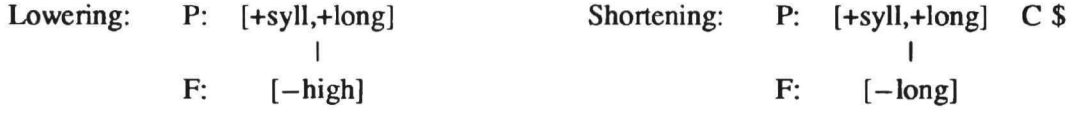
Clearly, epenthesis precedes vowel harmony. This is further supported by examples like [logiw-xa] 'let's pulverize', derived from /logw-xa/, in which the appearance of the epenthetic vowel blocks rounding from applying to the final vowel.

To summarize, the phonological system of Yawelmani offers an example of a very complex set of interactions between independently-motivated rules. The classical generative analysis of this data depends heavily on ordered application [4,5], but learning these rules would be easier in an architecture where they could apply simultaneously. Another factor complicating learning is that the environment of vowel harmony is only regular at the abstract underlying level; the alternations attributable to harmony are not completely systematic based solely on surface forms.

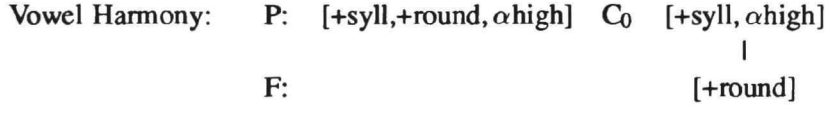
3. The "Many Maps" Architecture

As an alternative to standard generative analyses involving long lists of ordered rules, Lakoff [6,7] develops a theory of phonology which is essentially non-derivational in character. "Rules" are replaced by "constructions," which state well-formedness constraints within levels and correlations between levels. He recognizes three levels: a morphophonemic level (M), a phonemic level (P), and a phonetic level (F). Our attempts to actually implement Lakoff's work in a connectionist framework led to several revisions in the analysis of Yawelmani [11,14]. In this section we will describe our analysis, then turn to the question of learning this complex set of phonological rules in the next section.

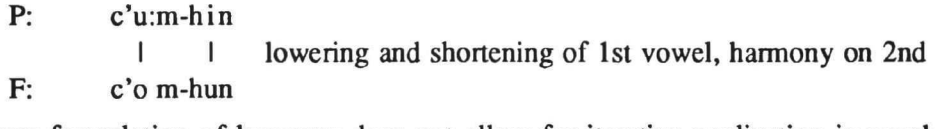
In a generative analysis, shortening comes *after* lowering because only long vowels can lower. In our model, both these processes are P-F constructions that apply simultaneously. Both have their environments satisfied at P-level and their changes realized at F-level. The rules are presented below using Lakoff's parallel mapping notation:



Vowel harmony is traditionally extrinsically ordered before both lowering and shortening, because of the restriction that vowels undergoing harmony must agree in height. If harmony followed lowering, there would be no rounding in the final vowel of examples like [c'omhun] 'destroys' (underlying form /c'u:m-hin/). But in our model, vowel harmony is another P-F construction. It applies simultaneously with the other two P-F rules; all three rules therefore have their environment at P. Ignoring iteration for the moment, the vowel harmony rule might be stated as:



An example of the interaction of these three constructions is given below.



The above formulation of harmony does not allow for iterative application in words like [t'ul-sut-hun] (from /t'ul-sit-hin/). In our model, harmony is actually stated as a P-F "cluster rule." Cluster rules use special circuitry described in [11,14] to process groups of segments undergoing the same change. The cluster rule for harmony first identifies the cluster "trigger" (a round vowel), and then marks as cluster "elements" all subsequent vowels in sequence that agree with it in height. The change sanctioned by the rule is applied to all these elements simultaneously. Our cluster rules are stated in a tabular format:

P-F Cluster Rule for Vowel Harmony:

Cluster type:	[+syllabic]
Trigger:	[+round, αhigh]
Element:	[αhigh]
Change:	[+round]

Our model also includes a syllabifier component that is part of the mapping from M-level to P-level [13]. Epenthesis and deletion processes in most languages are necessitated by the requirement that strings be

fully syllabified. In Yawelmani, epenthesis serves to break up consonant clusters which would otherwise be unsyllabifiable, because its syllable types are restricted to CV and CVC [5]. Since epenthesis is an M-P rule and harmony processes are always P-F, we correctly predict that the epenthetic vowel [i] will be visible at P-level, and thus undergo or block harmony, depending on the height of the preceding vowel.

The following examples illustrate the interactions of the four rules in our parallel formalism:

M:	du:ll-al		M:	du:l l-hin	
				|	epenthesis
P:	du:ll-al		P:	du:lil-hin	
	|	shortening		| | |	lowering, harmony
F:	doll-al	'might climb'	F:	do:lul-hun	'climbs'

With all three mutation rules, lowering, shortening, and harmony, stated as P-F rules, the problem of rule induction is considerably simplified, since there is no need to stipulate constraints on rule application. These are determined by the architecture of the model. We're now in a position to consider the problem of rule induction.

4. The Rule Learning Program

Our rule learning program takes as input a set of of M-level forms and their corresponding F-level realizations. Its output is a set of rules that collectively account for the differences between the two levels. We will discuss the process of learning the shortening rule to illustrate how mutation rules in general are acquired. Each rule accounts for a single change to a single segment. Rule environments are induced using Mitchell's version space technique [8]. Rules have two environments, a most specific environment (SPEC) and a most general (GEN). The SPEC is the intersection of all environments in which the rule has been seen to apply. Initially the SPEC is the string consisting of the changed segment of the first example seen, and up to three segments of context on either side. The GEN states the minimal conditions required for the rule to apply. Initially it contains just the major class features of the first input, i.e., sequences of [+cons] and [+syll], represented below as Cs and Vs. When the SPEC and GEN match, the rule has been refined completely.

Assume the first pair of inputs the learner examines is /do:shin/ and [doshin]. A comparison of the two strings reveals a difference in length of the initial vowel, so a prototype shortening rule is created. The shortening rule's initial SPEC is <d>o<s$hi>, where segments in angle brackets denote context, and the $ indicates a syllable boundary that was detected by the syllabifier. The /o:/ is not marked [+long] by convention in the SPEC, because this feature is the one changed by the rule. The rule's initial GEN is <C>V<CCV>. When shortening is seen to operate in other environments, the GEN will contract to the minimal common substring that characterizes all examples, namely <C>V<C>.

Suppose the second input happens to be /c'u:mhin/ and [c'omhun]. There are three differences in this string: the initial vowel lowers and shortens, and the final vowel rounds. The learner will hypothesize a separate rule for each change. Rules are indexed by the changes they cause, so the previously-formulated rule for shortening will be expected to account for this case as well. Further examples of shortening help to relax the SPEC by eliminating overly-restrictive features. After incorporating the /c'u:mhin/ example, the proto-rule's SPEC is <C>[V,+round]<C$hi>. And after processing the next pair, /bok'i:n/ and [bok'en], the SPEC becomes <C>V<C$>, and the GEN contracts to <C>V<C>.

Examples where shortening doesn't apply are also important, because they force the learner to make the GEN more specific. The underlying form /do:sal/ satisfies the GEN for shortening of the first vowel, but the surface form [do:sol] shows that shortening did not in fact happen. This causes the learner to look for some feature of the SPEC that is absent from the GEN and uniquely accounts for the rule's failure to apply. In this case, the only such feature is the coda marking of the final consonant. The GEN is updated to <C>V<C$>, which matches the SPEC, and the rule is now complete. The lowering rule is learned in a similar fashion.

When a sequence of vowels is observed to undergo the same change, a cluster rule is postulated. On the basis of examples such as /t'ul-sit-hin/, [t'ul-sut-hun], where the second and third high vowels undergo rounding, the learner creates a cluster rule for rounding. It generates single-segment GENs and SPECs for the trigger and element portions of the rule, and refines them as it processes additional examples, such as /do:s-al/, [do:s-ol]. The trigger GEN eventually settles on [+syll,+round], and the element GEN remains [+syll]. Unfortunately, these specifications do not explain certain cases where vowel harmony fails to apply, such as /do:s-hin/, which does not become *[dos-hun], but rather surfaces as [dos-hin].

The learner collects correlation statistics on trigger and element features in order to detect agreement relationships that would be expressed with α variables in conventional rules. The correlation matrix indicates that in all cases where harmony applies, triggers and elements agree in height, and in cases where harmony fails to apply, they disagree. Therefore the learner abandons the original harmony rule and generates a pair of rules. In one, the trigger and element specifications are both [+high], while in the other they are [−high]. We find this solution preferable to introducing variable binding for a feature [αhigh] into the clustering machinery. That would increase the complexity of the connectionist architecture that must implement these rules.

Although we are covering epenthesis last in this section, the learner is actually designed to detect and account for epenthesis and deletion phenomena first. They can generally be explained by syllabification, an M-P process. At this stage in the development of the model, the syllabifier relies on language-specific parameters which we supply for Yawelmani; it does not yet acquire these parameter values on its own. Learning the M-P epenthesis process in cases such as /ʔugn-hin/, [ʔugunhun] is simple and straightforward given the phonotactic constraints of Yawelmani. The syllabifier predicts an epenthetic vowel will appear in this example; all the rule learner needs to do is specify the quality of the vowel. The interaction of epenthesis with other rules follows from the architecture of the model. Notice that this automatically predicts that epenthesis (M-P) will feed harmony (P-F), as it does in the [ʔugunhun] example.

After epenthesis applies to the M-level form /ʔugn-hin/, its P-level representation is /ʔugIn-hin/. The /I/ is a [+high,−long] segment left unspecified for roundness, because the epenthetic vowel surfaces as either [i] or [u] depending on context. The P-level sequence then forms part of the data for learning the harmony rule. Thus we see that when learning P-F rules, the actual input strings must be obtained by applying previously-acquired M-P processes to the M-level strings, rather than looking at the M-level strings themselves. This also lets the model naturally account for the failure to round the final vowel in /logw-xa/. If the harmony rule looked at the underlying representation we would expect it to apply, but the epenthetic high vowel /I/ appearing at P-level in /logIw-xa/ blocks it, since it does not agree in height. This correctly predicts the surface form [logiw-xa], not *[logiw-xo].

As this section has shown, our rule learner has several components. One component learns SPECs and GENs for ordinary rules. Another learns the components of cluster rules: the cluster type, SPECs and GENs for the trigger and element, and the desired change. A third component recognizes insertion and

deletion phenomena that can be accounted for by epenthesis, and so do not need separate rules. Each component corresponds to a different piece of the connectionist architecture that implements these rules.

5. Discussion

Our treatment of vowel harmony is notably different from the proposals of Gasser & Lee [1] and Hare [3], both of which are based on properties of sequential recurrent networks. Their models are trained directly on surface sequences, and do not derive underlying representations to express regularities, although they do develop internal "hidden" representations. They focus on just harmony, and do not consider possible interactions of this process with other phonological rules. Hare's model suggests that a vowel will undergo harmony to become more like the trigger only if it is already sufficiently similar to the trigger. In Yawelmani the crucial similarity governing harmony is height, but the [αhigh] constraint applies only at the abstract underlying level. It is violated on the surface, because long high vowels serving as triggers or elements subsequently undergo lowering.

Insertions and deletions are always handled at M-P in our model (usually by the syllabifier), and mutations at P-F. This works fine for the examples we've tried from Yawelmani and other languages, and is suggestive of a more general pattern across languages. However, it remains an open question whether all languages fall into this pattern. If necessary, we can introduce additional mechanisms to allow rules to migrate between M-P and P-F to establish the correct feeding and blocking relations. We suspect such mechanisms will ultimately be needed.

Our model would be more impressive if it developed its own underlying representations from exposure to annotated surface forms alone. We see no reason why this cannot be done in principle, once a lexical component is added to provide the necessary information (syntactic or semantic) for recognizing allomorphs. We are exploring various possibilities.

The main result of our work to date is that rule learning can be made tractable by adopting a more cognitively natural (less sequential) formalism. Our M^3P architecture has another advantage: it is compatible with a connectionist implementation, and therefore does not violate any fundamental computational constraints associated with neural processing.

Acknowledgements

This research was supported by a contract from Hughes Research Laboratories, by the Office of Naval Research under contract number N00014-86-K-0678, and by National Science Foundation grant EET-8716324. We thank David Evans for helpful comments on an earlier draft of this paper.

References

[1] Gasser, M., and Lee, C.-D. (1989) Networks that learn phonology. Indiana University Computer Science Technical Report #300.

[2] Goldsmith, J. (to appear) Phonology as an intelligent system. To appear in a festschrift for Leila Gleitman, edited by D. Napoli and J. Kegl.

[3] Hare, M. (1989) The role of similarity in Hungarian vowel harmony: a connectionist account. Technical report, University of California at San Diego.

[4] Kenstowicz, M., and Kisseberth, C. (1979) *Generative Phonology: Description and Theory.* San Diego, CA: Academic Press.

[5] Kuroda, S.-Y. (1967) Yawelmani Phonology. Cambridge, MA: The MIT Press.

[6] Lakoff, G. (1988) A suggestion for a linguistics with connectionist foundations. In D. S. Touretzky, G. E. Hinton, and T. J. Sejnowski (eds.), Proceedings of the 1988 Connectionist Models Summer School, pp. 301-314. San Mateo, California: Morgan Kaufmann.

[7] Lakoff, G. (1989) Cognitive phonology. Draft of paper presented at the UC-Berkeley Workshop on Constraints vs. Rules, May 1989.

[8] Mitchell, T. M. (1978) Version Spaces. Doctoral dissertation, Stanford University. Available as technical report STAN-CS-78-711.

[9] Newman, S. (1944) Yokuts Language of California. New York: Viking Fund Publications in Anthropology.

[10] Touretzky, D. S. (1989) Toward a connectionist phonology: the "many maps" approach to sequence manipulation. Proceedings of the Eleventh Annual Conference of the Cognitive Science Society, pp. 188-195. Hillsdale, NJ: Erlbaum.

[11] Touretzky, D. S., and Wheeler, D. W. (1990) A computational basis for phonology. In D. S. Touretzky (ed.), Advances in Neural Information Processing Systems 2. San Mateo, California: Morgan Kaufmann.

[12] Touretzky, D. S., and Wheeler, D. W. (1990) Rationale for a "many maps" phonology machine. Proceedings of EMCSR-90: the Tenth European Meeting on Cybernetics and Systems Research. Vienna, Austria, April 1990.

[13] Touretzky, D. S., and Wheeler, D. W. (unpublished) Two derivations suffice: the role of syllabification in cognitive phonology. Manuscript draft.

[14] Wheeler, D. W., and Touretzky, D. S. (1989) A connectionist implementation of cognitive phonology. Technical Report CMU-CS-89-144, Carnegie Mellon University School of Computer Science. To appear in J. Goldsmith (ed.), Proceedings of the UC-Berkeley Workshop on Constraints vs, Rules in Phonology, University of Chicago Press.

Discovering Faithful 'Wickelfeature' Representations in a Connectionist Network

Michael C. Mozer

Department of Computer Science and
Institute of Cognitive Science
University of Colorado
Boulder, CO 80309-0430
e-mail: mozer@boulder.colorado.edu

Abstract. A challenging problem for connectionist models is the representation of varying-length sequences, e.g., the sequence of phonemes that compose a word. One representation that has been proposed involves encoding each sequence element with respect to its local context; this is known as a *Wickelfeature* representation. Handcrafted Wickelfeature representations suffer from a number of limitations, as pointed out by Pinker and Prince (1988). However, these limitations can be avoided if the representation is constructed with a priori knowledge of the set of possible sequences. This paper proposes a specialized connectionist network architecture and learning algorithm for the discovery of *faithful* Wickelfeature representations — ones that do not lose critical information about the sequence to be encoded. The architecture is applied to a simplified version of Rumelhart and McClelland's (1986) verb past-tense model.

A challenging problem for connectionist models is the manipulation and representation of varying-length sequences. Consider the problem of representing a sequence of symbols, say letters. Using symbolic, LISP-like structures, this is straighforward: A short string like **ARM** can be represented as **(A R M)**, and a longer string like **FIREARM** can be represented by concatenating extra symbols onto the list — **(F I R E A R M)**. However, using connectionist activity patterns to represent these strings is a more complex matter. The activity pattern must indicate not only what the symbols are, but their positions in the string. This suggests the straightforward idea of reserving a processing unit for for every possible symbol in every position, but this scheme requires knowing the maximum length of the sequence in advance. It also suffers from the serious difficulty that two sequences containing common subsequences may appear quite different. For example, using the notation *x/n* to refer to a unit that is activated by the symbol *x* in position *n*, activity patterns corresponding to **{ A/1, R/2, M/3 }** and **{ F/1, I/2, R/3, E/4, A/5, R/6, M/7 }** have no overlap. The common subsequence **ARM** is not represented by an overlap in the activity patterns because **ARM** appears in a different position in each string. Overlap between similar activity patterns is critical in connectionist representations because it determines how a connectionist network will generalize to novel instances: if a network responds a certain way to **ARM**, one might like it to respond similarly to **FIREARM**, yet the position-specific letter encoding will not facilitate this.

Any representation of sequences should satisfy four criteria.

- The representation must be *faithful* (Smolensky, 1987), meaning that a one-to-one mapping exists between sequences and activity patterns. This requirement may be relaxed somewhat in the context of a particular task: The representation need only be sufficient to perform the desired input-output mapping. If two sequences have exactly the same consequences in all situations, there is no need to encode them distinctly. Task-irrelevant features do not have to be captured in the representation.
- The representation must be capable of encoding sequences of varying lengths with a fixed number of units.
- The representation must be capable of encoding relationships between elements of a sequence.
- The representation should provide a natural basis for generalization. It is on this ground that the position-specific encoding fails.

Wickelgren (1969) has suggested a representational scheme that seems to satisfy these criteria and has been applied successfully in several connectionist models (Mozer, 1990; Rumelhart & McClelland, 1986; Seidenberg, 1990). The basic idea is to encode each element of a sequence with respect to its local context. For example, consider the phonetic encoding of a word. Wickelgren proposed context-sensitive phoneme units, each responding to a particular phoneme in the context of a particular predecessor and successor. I will call these units *Wickelphones*,

after the terminology of Rumelhart and McClelland. If the word *explain* had the phonetic spelling /eksplAn/, it would be composed of the Wickelphones $_{-}e_{k}$, $_{e}k_{s}$, $_{k}s_{p}$, $_{s}p_{l}$, $_{p}l_{A}$, $_{l}A_{n}$, and $_{A}n_{-}$ (where the dash indicates a word boundary). Assuming one Wickelphone unit for every such triple, activation of a word would correspond to a distributed pattern of activity over the Wickelphone units. With a fixed number of Wickelphone units, it is possible to represent arbitrary strings of varying length. Generally, this representation is faithful. In such cases, the unordered set of Wickelphones is sufficient to allow for the unambiguous reconstruction of the ordered string.

Rumelhart and McClelland devised a more compact and distributed encoding of phoneme sequences that depended on *features* of the phonemes rather than the phonemes themselves. Units in this *Wickelfeature* representation encode triples of phonemic features (such a "voiced" or "dental"). Smolensky (1987) provides a formalism that allows the Wickelphone and Wickelfeature encodings to be viewed in a uniform representational framework, as tensor products of feature vectors. In the remainder of this paper, I use the term "Wickelfeature" to denote a context-sensitive encoding of features of sequence elements or of the elements themselves, thereby subsuming the term "Wickelphone" and allowing the representation to be applied to arbitrary sequences.

Pinker and Prince (1988; Prince & Pinker, 1988) point to several serious limitations of handcrafted Wickelfeature representations, in particular the representation used by Rumelhart and McClelland in their model of learning past tenses of English verbs. One critical limitation is that if the class of sequences to be represented contain repeated subsequences of length two or more, the resulting representation is ambiguous. For example, the set of Wickelfeatures { $_{-}A_{B}$, $_{A}B_{A}$, $_{B}A_{B}$, $_{A}B_{-}$ } could correspond to the sequence **ABAB** or to **ABABAB** or an infinite number of other such strings. Similarly, the set { $_{-}A_{B}$, $_{A}B_{X}$, $_{B}X_{A}$, $_{X}A_{B}$, $_{A}B_{Y}$, $_{B}Y_{A}$, $_{Y}A_{B}$, $_{A}B_{-}$ } could correspond either to sequence **ABXABYAB** or **ABYABXAB**. Thus, the Wickelfeature representation can lose order information.

There are potential ways around these problems. One quick solution is to represent more contextual information in the Wickelfeatures. If a Wickelfeature consists of v sequence elements rather than just three, confusions arise only if the strings contain repeated subsequences of length $v-1$ or greater. As v grows, however, the representation becomes more and more localist and loses the advantages that we set out to attain. Another solution is to have the Wickelfeature units be activated in a graded fashion, not all-or-nothing. This would allow a unit to signal the number of instances of that Wickelfeature in a sequence, which handles the **ABAB** problem. Alternatively, the amount of activity could correspond to the position in a sequence; in the **ABXABYAB** example, the $_{A}B_{X}$ unit could be less active than the $_{A}B_{Y}$ unit, indicating its primacy in the sequence.

Hand coding Wickelfeature representations of this sort gets quite tricky. In this paper, I report on an alternative approach using connectionist learning algorithms to discover Wickelfeature-like representations. The advantage of leaving the job to learning is that whatever representations the system develops, they are assured of being sufficient for the domain at hand, i.e., they will satisfy the faithfulness criterion mentioned above. A further advantage of using learning is that by discovering only domain-relevant Wickelfeatures, the overall representation can be more compact. For instance, a system whose task is to encode English letter strings as Wickelfeatures will not develop a $_{p}K_{T}$ unit.

A network architecture to learn Wickelfeatures

The approach I have taken involves training a network to map input sequences to target output patterns through a layer of units that learn to respond as Wickelfeatures. It does not much matter what the output patterns are; they could be localist representations of the sequences, responses to the sequence, or perhaps sequences themselves. Figure 1 shows a schematic drawing of an architecture that performs this mapping. The input layer represents a small window on the sequence. At any time, the input layer views several consecutive elements of the sequence — three elements in the Figure. Presenting a complete sequence to the network involves sliding the sequence through the window. More concretely, time is quantized into discrete *steps*, and at each time step, the sequence is advanced by one position in the input window. Once the entire sequence has been presented, the output units should respond appropriately. The output layer is activated by the context layer, the purpose of which is to remember those elements of the input sequence that are critical for performing the input-output mapping. At each time step, units in the context layer integrate their current values with the new input to form a new context representation.

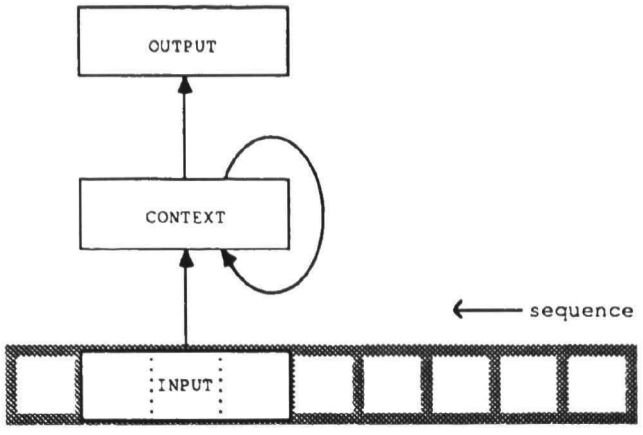


Figure 1. A three-layered recurrent network consisting of input, context, and output units. Each labeled box indicates a set of processing units. The arrows indicate complete connectivity from one layer to another.

The context layer thus forms a *static* internal representation of the dynamic input sequence. The goal of learning, of course, is for this to become a Wickelfeature representation. The use of a sliding input window does part of the job: At each time, the context units can only see a local "chunk" of the sequence. This allows the context units to detect local conjunctions of sequence elements, or conjunctions of features of sequence elements. Once activated by a pattern in the input, the context units should remain on. Thus, it seems sensible to have self-connected context units, but not to connect each context unit to each other, using an activation function like:

$$c_i(t+1) = d_i c_i(t) + s[net_i(t)] ,$$

where $c_i(t)$ is the activity level of context unit i at time t, d_i is a decay weight associated with the unit, s is a sigmoid squashing function, and $net_i(t)$ is the net input to the unit:

$$net_i(t) = \sum_j w_{ji} x_j(t) ,$$

$x_j(t)$ being the activity of input unit j at time t, w_{ji} the connection strength from input unit j to context unit i. Thus, a context unit adds its current activity, weighted by the decay factor, to the new input at each time. The decay factor allows old information to fade over time if d_i is less than one.

To summarize, the Wickelfeature-learning architecture differs from a generic recurrent architecture for temporal sequence recognition in three respects: (1) the input layer consists of a small temporal window holding several elements of the input sequence; (2) connectivity in the context layer is restricted to one-to-one recurrent connections; and (3) integration over time in the context layer is linear.

A training algorithm for the Wickelfeature architecture

The standard procedure for training a recurrent network with temporally-varying inputs using the back-propagation algorithm is to "unfold" the network in time (Rumelhart, Hinton, & Williams, 1986), transforming the recurrent network into a feedforward network. The unfolding procedure requires that each unit remember a temporal history of its activation values and is computation intensive. For the Wickelfeature architecture, however, the unfolding procedure can be avoided, as described by Mozer (1989). To summarize the result, consider a sequence with s elements and an architecture with a v-element window. The number of time steps, t, required to slide the sequence through the window is simply $s-v+1$. Consequently, at time t the network receives a target vector over the output units and an error E can be computed. Using the ordinary back propagation procedure, the error derivative with respect to each context unit i,

$$\delta_i(t) = \frac{\partial E}{\partial c_i(t)}$$

can be computed. The weight update rule for the recurrent connections, d_i, is then:

$$\Delta d_i = -\varepsilon \delta_i(t)\alpha_i(t)\,,$$

where $\alpha_i(t)$ is defined by the recurrence relation

$$\alpha_i(\tau) = c_i(\tau-1) + d_i\,\alpha_i(\tau-1)$$

with boundary value $\alpha_i(0) = 0$. Similarly, the weight update rule for the input-context connections, w_{ji}, is:

$$\Delta w_{ji} = -\varepsilon \delta_i(t)\beta_{ji}(t)\,,$$

where $\beta_{ji}(0) = 0$ and

$$\beta_{ji}(\tau) = s'[net_i(\tau)]x_j(\tau) + d_i\beta_{ji}(\tau-1)\,.$$

Thus, explicit back propagation in time is not necessary. Bachrach (1988), Gori, Bengio, and De Mori (1989), and Williams and Zipser (1989) have independently discovered the idea of computing an activity trace during the forward pass as an alternative to back propagation in time. However, this is the first use of the architecture for the purpose of learning Wickelfeature-like representations.

Simulation results

Implementation details

In the simulations to be reported, an additional parameter z_i — called the *zero point*, was added to the context-unit activation function, for reasons described by Mozer (1989). The complete activation function is:

$$c_i(t+1) = d_i\,c_i(t) + s[net_i(t)] + z_i\,,$$

where the value of z_i is determined by gradient descent as for the other parameters.

The initial input-context and context-output connection strengths were randomly picked from a zero-mean gaussian distribution and normalized such that the L1 norm of the fan-in (incoming) weight vector was 2.0. The z_i were initially set to -0.5, and the d_i picked from a uniform distribution over the interval .99-1.01. The weights were updated only after a complete presentation of the training set (an *epoch*). Momentum was not used. Learning rates were adjusted dynamically for each set of connections according to a heuristic described by Mozer (1989).

Learning Wickelfeatures

Starting with a simple example, the network was trained to identify four sequences: **_DEAR_**, **_DEAN_**, **_BEAR_**, and **_BEAN_**. Each symbol corresponds to a single sequence element and was represented by a random binary activity pattern over three units. The input layer was a two-element buffer through which the sequence was passed. For **_DEAR_**, the input on successive time steps consisted of **_D**, **DE**, **EA**, **AR**, **R_**. The input layer had six units, the context layer two, and the output layer four. The network's task was to associate each sequence with a corresponding output unit. To perform this task, the network must learn to discriminate **D** from **B** in the first letter position and **N** from **R** in the fourth letter position. This can be achieved if the context units learn to behave as Wickelfeature detectors. For example, a context unit that responds to the Wickelfeatures **_D** or **DE** serves as a **B-D** discriminator; a unit that responds to **R_** or **AR** serves as an **N-R** discriminator. Thus, a solution can be obtained with two context units.

Fifty replications of the simulation were run with different initial weights. The task was learned in a median of 488 training epochs, the criterion for a correct response being that the output unit with the largest value was the appropriate one. Figure 2 shows the result of one run. The weights are grouped by connection type, with the input-context connections in the upper-left array, followed by the decay connections (d_i), zero points (z_i), and context-output connections. Each connection is depicted as a square whose *area* indicates the relative weight magnitude, and shading the weight sign — black is positive, white is negative. The sizes of the squares are normalized within each array such that the largest square has sides whose length is equal to that of the vertical bars on the right edge of the array. The absolute magnitude of the largest weight is indicated by the number in the upper-right corner. Because normalization is performed within each array, weight magnitudes of different connection types must be

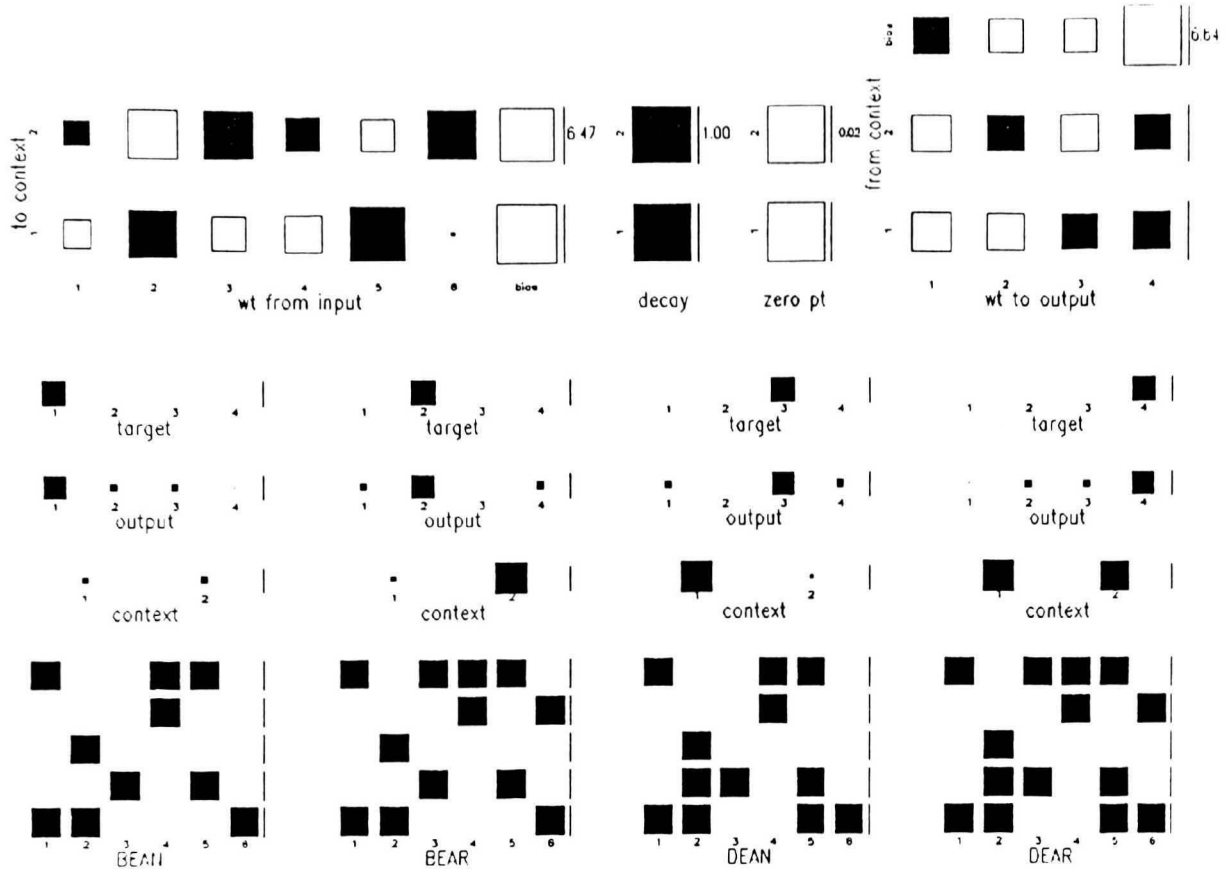


Figure 2. The DEAR/DEAN/BEAR/BEAN problem. The upper half of the figure shows learned weights in the network, the lower half activity levels in response to each of the four input sequences.

compared with reference to the normalization factors. The units within each layer are numbered. The weights feeding into and out of context unit 1 have been arranged along a single row, and the weights of context unit 2 in the row above. Bias terms (i.e., weight lines with a fixed input of 1.0) are also shown for the context and output units.

For the activity levels in the lower half of the figure, there are four columns of values, one for each sequence. The input pattern itself is shown in the lowest array. Time is represented along the vertical dimension, with the first time step at the bottom and each succeeding one above the previous. The input at each time reflects the buffer contents. Because the buffer holds two sequence elements, note that the second element in the buffer at one time step (the activity pattern in input units 4-6) is the same as the first element of the buffer at the next (input units 1-3). Above the input pattern are, respectively, the context unit activity levels after presentation of the final sequence element, the output unit activity levels at this time, and the target output values. The activity level of a unit is proportional to the area of its corresponding square. If a unit has an activity level of 0, its square has no area — an empty space. The squares are normalized such that a "unit square" — a square whose edge is the length of one of the vertical bars — corresponds to an activity level of 1. While the input, output, and target activity levels range from 0 to 1, the context activity levels can lie outside these bounds, and are, in fact, occasionally greater than 1.

With these preliminaries out of the way, consider what the network has learned. At the completion of each sequence, the context unit activity pattern is essentially binary. Context unit 1 is off for **_BEAN_** and **_BEAR_**, and on for **_DEAN_** and **_DEAR_**; thus, it discriminates **B** and **D**. Context unit 2 is off for **_BEAN_** and **_DEAN_**, and on for **_BEAR_** and **_DEAR_**; thus it discriminates **N** and **R**. However, the context units do not behave in a straightforward way as Wickelfeatures. If context unit 1 were sharply tuned to, say, **_D**, the input-context weights should serve as a matched filter to the input pattern **_D**. This is not the case: the weights have signs -+--+- but the **_D** input pattern is 110011. Nor is context unit 1 tuned to the **DE**, whose input pattern is 011010. Instead, the unit appears to be tuned equally to both patterns. By examining the activity of the unit over time, it can be determined that the unit

is activated partly by **_D** and partly by **DE** but by no other input pattern. This makes sense: **_D** and **DE** are equally valid cues to the sequence identity, and as such, evidence from each should contribute to the response. To get a feel for why the detector responds as it does, note that **_D** (110011) is distinguished from **_B** (110001) by activity in unit 5; **DE** (011010) from **BE** (001010) by activity in unit 2. The weights from inputs 2 and 5 to context unit 1 are positive, allowing the unit to detect **D** in either context. The other weights are set so as to prevent the unit from responding to other possible inputs. Thus, the unit selects out key features of the Wickelfeatures **_D** and **DE** that are not found in other Wickelfeatures. As such, it behaves as a **_DE** Wickelfeature detector, and context unit 2 similarly as a **AR_** detector.

Generalization testing supports the notion that the context units have become sensitive to these Wickelfeatures. If the input elements are permuted to produce sequences like **AR_BE**, which preserves the Wickelfeatures **AR_** and **_BE**, context unit responses are similar to those of the original sequences. However, with permutations like **_RB_**, **_DAER_**, and **DEAR** (without the end delimiters), which destroy the Wickelfeatures **AR_** and **_BE**, context unit responses are not contingent upon the **D**, **B**, **N**, and **R**. Thus, the context units are responding to these key letters, but in a context-dependent manner.

Learning the regularities of verb past tense

In English, the past tense of many verbs is formed according to a simple rule. Regular verbs can be divided into three classes, depending on whether the past tense is formed by adding /ˆd/ (an "ud" sound), /t/, or /d/. Examples of the classes are /dEpend/ (*depend*), /fAs/ (*face*), and /dEscrIb/ (*describe*), respectively. Each string denotes the phonetic encoding of the verb in italics, and each symbol a single phoneme. The phoneme notation and the examples have been borrowed from Rumelhart and McClelland (1986). The rule for determining the class of a regular verb is as follows.

> If the final phoneme is dental (/d/ or /t/), add /ˆd/;
> else if the final phoneme is an unvoiced consonant, add /t/;
> else (the final phoneme is voiced), add /d/.

A network was trained to classify the twenty examples of each class. Each phoneme was encoded by a set of four trinary acoustic features (see Rumelhart & McClelland, 1986, Table 5). The input layer of the network was a two-element buffer, so a verb like /kamp/ appeared in the buffer over time as **_k, ka, am, mp, p_**. The underscore is a delimiter symbol placed at the beginning and end of each string. The network had eight input units (two time slices each consisting of four features), two context units, and three output units — one for each verb class.

In fifteen replications of the simulation, the network performed at 90% within 100 epochs, learned the training set perfectly in under 1000 epochs. A verb was considered to have been categorized correctly if the most active output unit specified the verb's class. The network has learned the underlying rule, as evidenced by perfect generalization to novel verbs. Typical weights learned by the network are presented in Figure 3, along with the output levels of the two context units in response to twenty verbs. These verbs, though not part of the training set, were all classified correctly.

The response of the context units is straightforward. Context unit 1 has a positive activity level if the final phoneme is a dental (/d/ or /t/), negative otherwise. Context unit 2 has positive activity if the final phoneme is unvoiced, near zero otherwise. These are precisely the features required to discriminate among the three regular verb classes. In fact, the classification rule for regular verbs can be observed in the context-output weights (the rightmost weight matrix in Figure 3). Connections are such that output unit 1, which represents the "add /ˆd/" class, is activated by a final dental phoneme; output unit 2, which represents the "add /t/" class, is activated by a final non-dental unvoiced phoneme; and output unit 3, which represents "add /d/" class, is activated by a final non-dental voiced phoneme.

Note that the decay weights in this simulation are small in magnitude; the largest is .02. Consequently, context units retain no history of past events, which is quite sensible because only the final phoneme determines the verb class. This fact makes verb classification a simple task: it is not necessary for the context units to hold on to information over time. Simulations were also conducted giving the network the same verb classification task, but reversing the order of the phonemes; instead of /dEpend/, /dnepEd/ was presented. In this problem, the relevant

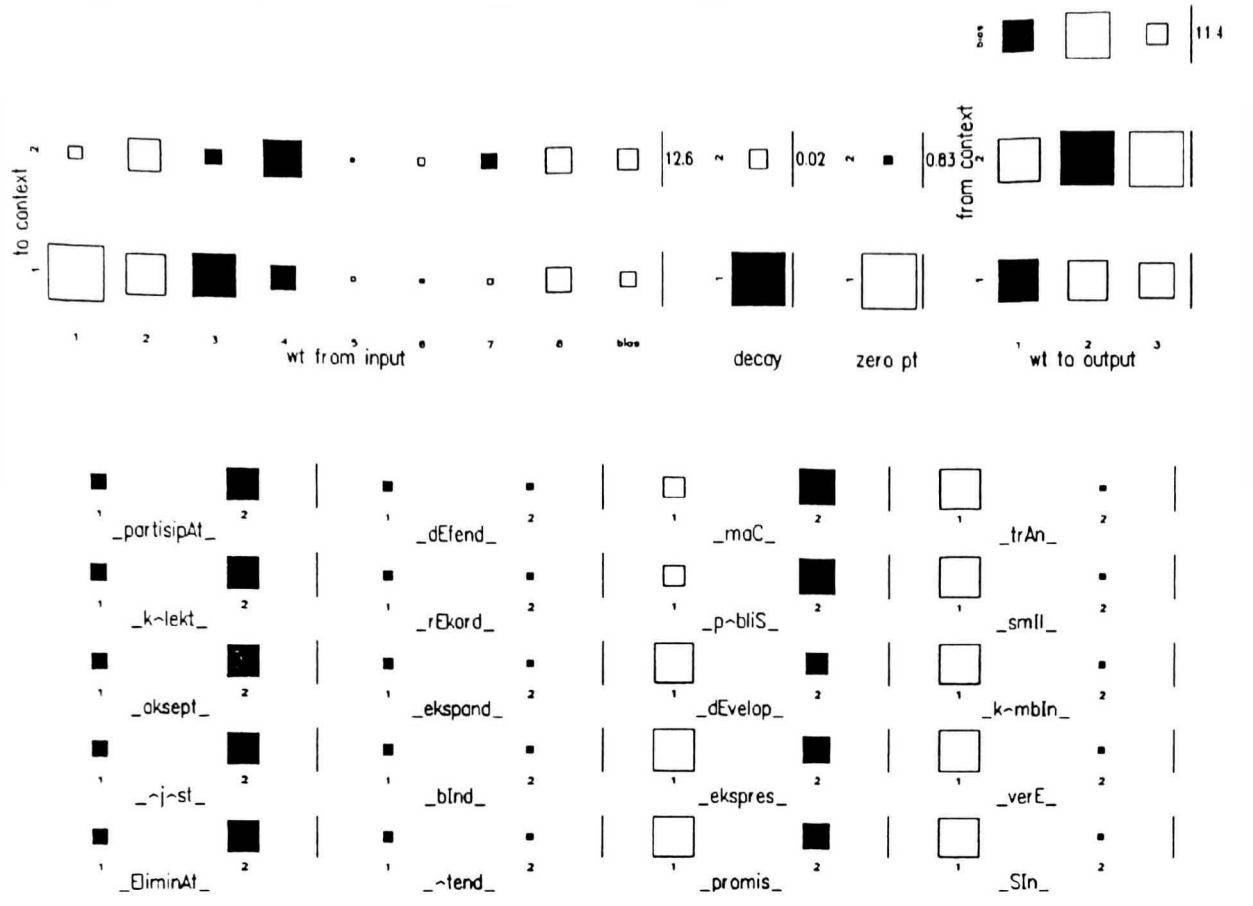


Figure 3. The regular verb problem. The upper half shows learned weights in the network, the lower half shows the final activity levels of the context units in response to a variety of verbs. Verbs in the first column all end with /t/, in the second column with /d/, in the third column with an unvoiced consonant, and the fourth column with a voiced consonant or vowel.

information comes at the *start* of the sequence and must be retained until the sequence is completed. Nonetheless, the network is able to learn the task. Interestingly, a more standard network architecture was unsuccessful at learning the task.

Large verb simulation

To study a more difficult task, the regular-verb categorization problem was extended to a larger corpus of verbs. As before, the task was to classify each verb according to the manner in which its past tense is formed. The complexity of the task was increased by including both regular and irregular verbs, 136 training instances altogether, and a total of thirteen response categories — three for regular forms and ten for irregular (see Mozer, 1989, for examples of these categories). The categories are based loosely on a set suggested by Bybee and Slobin (1982).

The corpus of verbs was borrowed from the Rumelhart and McClelland (1986) model. The model, designed to account for children's acquisition of verb past tenses, produces the past tense of a verb given its infinitive form as input. The representation used at both input and output ends is a handcrafted Wickelfeature encoding of the verb, built into the model. The purpose of this simulation is to demonstrate that a network, given a sequence of phonemes, can *learn* a representation like that presupposed by Rumelhart and McClelland's model.

The task is difficult. The verb classes contain some internal regularities, but these regularities are too weak to be used to uniquely classify a verb. For instance, all verbs in category 3 end in a /d/ or /t/, but so do verbs in categories 4, 5, and 11. Whether a verb ending in /d/ or /t/ belongs in category 3 or one of the other categories depends on whether it is regular, but there are no simple features signaling this fact. Further, fine discriminations are necessary because two outwardly similar verbs can be classified into different categories. *Swim* and *sing* belong to category 10, but *swing* to category 12; *ring* belongs to category 10, but *bring* to category 8; *set* belongs to

category 4, but *get* to category 11. Because the category to which a verb belongs is somewhat arbitrary, the network must memorize a large number of special cases.

The network architecture was similar to that used in the regular verb example. The input layer was a two-phoneme buffer, and the encoding of phonemes was the same as before. The output layer consisted of thirteen units, one for each verb class, and the context layer contained 25 units.

In ten replications of the simulation, the network learned to select the correct category in about 500 epochs. At intermediate stages of learning, verbs are sometimes "overregularized", as when the past tense of *eat* was considered to be *eated*. Overgeneralization occurs in other respects, as when *sit* was misclassified in the category of verbs whose past tense is the same as the root — presumably by analogy to *hit* and *fit* and *set*. Interpretation of the behavior of individual context units is difficult, but by examining similar input sequences that are classified differently, e.g., /riN/ and /briN/, one can pinpoint context units responsible for certain behaviors.

These simulations demonstrate the feasibility of constructing faithful Wickelfeature-like representations using connectionist learning procedures, instead of having to craft the representations by hand. Further, the simulations show that intrinsically temporal or sequential input can be dealt with as such, instead of as static patterns. This is a necessary first step in the modeling of language and speech processes.

Acknowledgements

Thanks to Jeff Elman and Yoshiro Miyata for their insightful comments and assistance. The graphical displays of network states are due to Miyata's SunNet simulator. Dave Rumelhart and Jay McClelland were kind enough to provide me with the phonological encoding and classification of verbs from their simulation work. This research was supported by Contracts N00014-85-K-0450 NR 667-548 and N00014-85-K-0076 with the Office of Naval Research, a grant from the System Development Foundation, and a Junior Faculty Development Award from the University of Colorado.

References

Bachrach, J. (1988). *Learning to represent state.* Unpublished master's thesis, University of Massachusetts, Amherst.

Bybee, J. L., & Slobin, D. I. (1982). Rules and schemas in the development and use of the English past tense. *Language, 58,* 265-289.

Gori, M., Bengio, Y., & Mori, R. de (1989). BPS: A learning algorithm for capturing the dynamic nature of speech. In *Proceedings of the First International Joint Conference on Neural Networks, Volume 2* (pp. 417-423).

Mozer, M. C. (1989). A focused back-propagation algorithm for temporal pattern recognition. *Complex Systems, 3.*

Mozer, M. C. (1990). In *The perception of multiple objects: A connectionist approach.* Cambridge, MA: MIT Press/Bradford Books.

Pinker, S., & Prince, A. (1988). On language and connectionism. *Cognition, 28,* 73-193.

Prince, A., & Pinker, S. (1988). Wickelphone ambiguity. *Cognition, 30,* 189-190.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart & J. L. McClelland (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Volume I: Foundations* (pp. 318-362). Cambridge, MA: MIT Press/Bradford Books.

Rumelhart, D. E., & McClelland, J. L. (1986). On learning the past tenses of English verbs. In J. L. McClelland & D. E. Rumelhart (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Volume II: Psychological and biological models* (pp. 216-271). Cambridge, MA: MIT Press/Bradford Books.

Seidenberg, M. S. (1990). Word recognition and naming: A computational model and its implications. In W. D. Marslen-Wilson (Ed.), *Lexical representation and process.* Cambridge, MA: MIT Press.

Wickelgren, W. (1969). Context-sensitive coding, associative memory, and serial order in (speech) behavior. *Psychological Review, 76,* 1-15.

Williams, R. J., & Zipser, D. (1989). Experimental analysis of the real-time recurrent learning algorithm. *Connection Science, 1,* 87-111.

Constraints on Assimilation in Vowel Harmony Languages

Mary Hare
UC San Diego

Abstract

Over the last 10 years, the assimilation process referred to as vowel harmony has served as a test case for a number of proposals in phonological theory. Current autosegmental approaches successfully capture the intuition that vowel harmony is a dynamic process involving the interaction of a sequence of vowels; still, no theoretical analysis has offered a non-stipulative account of the inconsistent behavior of the so-called "transparent", or disharmonic, segments.

The current paper proposes a connectionist processing account of the vowel harmony phenomenon, using data from Hungarian. The strength of this account is that it demonstrates that the same general principle of assimilation which underlies the behavior of the "harmonic" forms accounts as well for the apparently exceptional "transparent" cases, without stipulation.

I. Introduction[1]

The current paper proposes a connectionist processing account of certain aspects of vowel harmony in Hungarian. The paper has two interrelated goals. First, it offers an explanatory account of the behavior of the so-called *transparent* vowels in that language. Second, this account relies crucially on a connectionist theory of sequential processes: thus to the extent it succeeds it demonstrates the utility of connectionist models as an explanatory tool in the study of linguistic phenomena.

The paper is organized in the following manner: I first review some facts about the vowel harmony process in Hungarian which present difficulties to analysis. Second, I introduce the model of sequential processes developed by Jordan (1986). The core of the paper then involves a series of parametric studies, whose aim is to determine the conditions on assimilation in a network of this type. Having established what factors constrain assimilation in the sequential network, I return to the Hungarian data, and show that the interaction of these same factors predicts the correct pattern of behavior for both harmonic and transparent vowels in that language.

II. The data

The Hungarian vowel system is as shown below. This is a seven vowel system, and each vowel has a long counterpart which is phonemic. Notice that there is a round - nonround distinction among the non-low vowels, and that while /e:/ is a mid vowel, /e/ is analyzed as being low.

front:		back:
unrounded:	rounded:	
i i:	ü ü:	u u:
e:	ö ö:	o o:
e		a

Hungarian exhibits front-back harmony: in general roots contain only front or only back vowels, and suffix vowels alternate to agree in backness with those of the root. The following data exemplify the harmony phenomenon. These are examples of consistent front- and back-vowel roots, followed by the dative suffix. Note that after a front root, the suffix takes the form *nek* while after a back root the same suffix is realized as *nak* (data from Vago 1979).

(1) Front roots:

iker 'twin'	iker-nek	'twin-DAT'
tükör 'mirror'	tükör-nek	'mirror-DAT'

(2) Back roots:

varos 'city'	varos-nak 'city-DAT'
kapu 'gate'	kapu-nak 'gate-DAT'

There are a number of exceptions to the pattern of consistent harmony within roots. The most important exception involves the class of non-low front unrounded vowels (/i/, /i:/, /e:/, and on some accounts /e/). As the following examples demonstrate, these can also appear in the same root as a back vowel. In these cases the back vowel of the root, regardless of its position, determines the backness quality of the suffix vowel.

(3)

bika 'bull'	bika-nak 'bull-DAT'
izom 'tendon'	izom-nak 'tendon-DAT'

(4)

kosci 'carriage'	kosci-nak 'carriage-DAT'
taxi	taxi-nak 'taxi-DAT'

Note, however, that in certain environments the front vowels *do* determine the backness value of the suffix. This is the case if the root contains only front vowels, as in (1), or if the root ends in a sequence of such vowels, as in the (borrowed) forms shown below (data from Kontra and Ringen 1987).

(5)

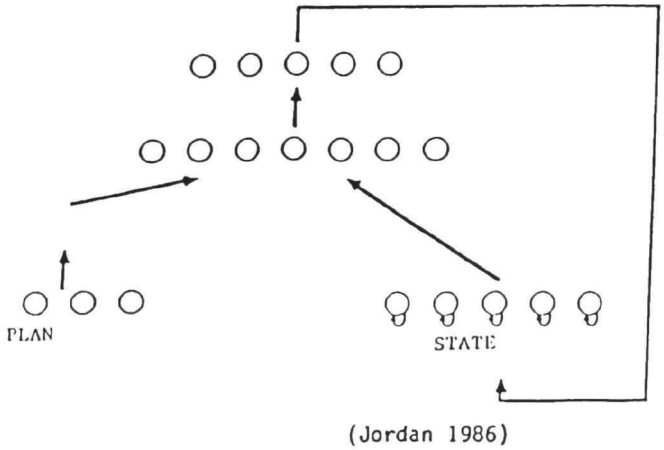
aspirin	aspirin-nek
	* aspirin-nak
	bronkitis
	* bronkitis-nak

The problem, then, is that an identical vowel may behave harmonically in one environment, while violating harmony in another. This complication has often been dealt with in the literature by positing a number of different sources for the segment in question, and allowing the harmonic-nonharmonic distinction to follow from this (Clements 1986, van der Hulst 1985, Ringen 1989, among others). One drawback to such an approach is that there is no reason to establish these differences in derivational source except to distinguish between harmonic and nonharmonic behavior: there is no other behavior in the phonology of the language that motivates it. A preferable account would be one in which these differences in behavior follow from general conditions on the model. In what follows I will use a connectionist model of assimilation to suggest one such account.

III. Sequential processing in the connectionist framework

The account that is being developed here relies on the theory of sequential processes developed by Michael Jordan. Jordan (1986) describes an interesting series of models of coarticulation effects, using a recurrent connectionist network which learns to produce an ordered sequence of output patterns in response to a given input. The network is illustrated in Diagram I.

(6) Diagram I



(Jordan 1986)

These models involve at least three layers of processing units: input, internal (or *hidden*), and output. Activation passes from the input to the output along weighted connections. Input to the model consists of two parts. The first, labeled the *plan*, is an arbitrary vector that triggers the production of a given sequence. In addition, the *state* of the system (that is, the current output) is fed back over fixed connections and constitutes part of the input at the next cycle. This serves as a temporal context and aids the system in learning what part of the sequence is the next to be produced. Learning is accomplished through the back propagation of error algorithm. After each input is presented, the output that results is compared to the desired output, and the discrepancy between the two is computed. This discrepancy is the *error* on that pattern. The weights on the connections are modified slightly to minimize this error. This process is repeated until some criterion of acceptability is reached.

In the simulations to be discussed here, as in Jordan's coarticulation model, output at any given time consists of a single phoneme represented by a vector corresponding to a distinctive feature description. A word or other longer sequence is represented over time as a string of phonemes on successive output cycles. One interesting property of this network is that particular features of a phoneme can be left unspecified for any value. This is accomplished by having no error signal propagated back from that unit. Instead of learning to match a particular teacher, the unit picks up its specification from some other pattern in the sequence.

In what follows I will use the term *assimilation* to refer to the tendency of an unspecified output unit (hereafter a *don't care* unit) to take on a value influenced by one of its neighbors in time. Jordan shows that outputs tend to follow as smooth a trajectory as possible: thus a don't care unit might be expected to assimilate most strongly to its immediate temporal predecessor. In certain cases, however, the don't care unit ignores its immediate predecessor, and takes on a value close to that of an earlier pattern in the sequence. The question, then, is to determine what factors influence the choice of assimilatory trigger. Note that this exactly the problem in the Hungarian data, as well.

IV. Conditions on assimilation in the sequential network[2]

The following set of simulations were designed to test the hypothesis that the similarity between vowels (in a sense which will be made precise below) is a crucial factor in determining the choice of assimilatory trigger. Stimuli were output sequences as in the example below.

(7)

1 0100010
2 1011101
3 *100010

Each output was a seven-bit distributed pattern, and for each plan the network learned to produce a three-pattern sequence whose members will be referred to as 1, 2, and 3. In each sequence the first two patterns (1 and 2) are specified for all seven units, while the third (3) has one don't care unit, in initial position in the string. (The don't care unit is indicated by the asterisk.) 1 and 2 have opposing values on this first unit.

Sets of patterns were devised in which the final two lines (2 & 3) were held constant with certain number of units in common. This measure of 'units in common' is referred to as the 'hamming distance' between 2 and 3, and is a measure of vector similarity. The first line in the sequence (line 1) was varied in similarity to the other two by manipulating the hamming distance between them This was done in the following way. The pattern given in (7) was the first 3-line sequence in one such set. Here 2 & 3 have opposing values on the last six bits, while 1 and 3 are identical. In the second sequence of this set (example 8) 2 and 3 remain unchanged while 1 is varied to differ from 3 on one unit. In the third sequence, (given in 9) 1 and 3 differ on two units.

(8)

1 0000010
2 1011101
3 *100010

(9)

1 0010010
2 1011101
3 *100010

This process was repeated, steadily decreasing the similarity between 1 and 3 until the set consisted of seven 3-line sequences. Note that by *similarity* I am speaking of hamming distance, a measure of overall vector similarity, and not simply the presence of similar values on any single unit.

a. Training

These sequences served as teaching output to the network described above. This network was trained on each sequence for 2000 iterations, where an iteration is one presentation of one pattern. In learning to produce the sequences, the network also assigns a value to the don't care unit. This unit is expected to simply maintain the value of the previous pattern; the goal of these simulations is to determine under what conditions the *don't care* unit reverts to the value of the first pattern instead. After 2000 iterations, the training was stopped and the actual output was examined to determine the value taht the don't care unit had taken on.

b. Results

Results from the first set of simulations show that the don't care unit in 3 consistently assimilates to the corresponding unit in 1 when these two patterns are most similar. Thus although the default case in the Jordan network is for a don't care unit to maintain the value of the immediately previous output, this unit does not exclusively influence the result. If the pattern two time steps back is strongly similar to the target, the don't care unit will take a value nearer the corresponding unit in that pattern instead.

These results are given in Graph I, which should be read as follows. Distance along the x-axis measure the similarity between patterns 1 and 3 - that is, between the first and third line of each sequence. The y-axis gives the activation level taken on by the don't care unit after 2000 iterations of learning. The boldface bar gives the average activation over a number of trials, while the line through each bar marks the limits of the variability. As the graph shows, as patterns 1 and 3 become more alike, there is a corresponding increase in the influence of 1 as assimilatory trigger.

These results are typical of a pattern which emerged over a number of pattern sets. The second graph gives the results from a second set of sequences, in which the second pattern was held constant at a hamming distance of 5 units from the target, while the first pattern was varied as before. The only difference between the simulations reported above and this set is that 2, the second pattern in the sequence, is here slightly more similar to 3, the target. In this set, the same pattern of results emerges.

Notice also that the second graph begins with an additional column, where all values are clustered near 0, indicating low assimilation to 1 and high assimilation to 2. This is the output from what will be referred to at the identity condition, where 1 and 2 are not only equal in hamming distance from 3, but are identical. This result shows that when the two potential trigger patterns are identical or nearly so, the target pattern assimilates to the second of the two in all cases. This in not surprising, given the model. A basic property of these networks is that, similar inputs produce similar outputs. Since here temporal context is treated as part of the input, patterns learned in very similar temporal contexts are expected to exhibit very similar behavior.

These simulations were repeated under a number of conditions, with the hamming distance between 2 and 3 progressively decreased. The same pattern of results continued to appear, although in an increasingly attenuated form. Consistently, the influence of the first pattern of the sequence is strongest when it is most similar to the target.

V. Analysis of Hungarian vowel harmony

This pattern of results shows that in a processor of this sort the similarity structure of output strings across time influences assimilatory behavior. Returning to the Hungarian data, let us consider how the facts of the transparent vowels of that language agree with the behavior of the sequential network.

Here I modeled the behavior of a series of Hungarian words in the same assimilation task. In this case the output sequences were not arbitrary bit strings chosen only for their similarity structure, but vectors corresponding to distinctive feature representations of phonemes. The features used to represent the vowels were *back*, *high*, *low*, and *round*.

(10) Vowel Code:

i	0100	ö	0001
e:	0000	u	1101
e	0010	o	1001
ü	0101	a	1010

Words being modeled were represented only by their vowels. Each sequence consisted of vectors representing two or more root vowels specified for all features, and a third which represented the vowel of the dative suffix. This was given as a low unrounded vowel unspecified for [back]. As before, lack of specification equated to a don't care condition on the relevant unit.

(11) iker - nek

i	0100
e	0010
e	*010

Each pattern was learned separately, as before, for 2000 iterations. At this point the underspecified vowel had taken on a value for [back] influenced either by its immediate predecessor, or by an earlier member of the sequence.

To summarize the results of the earlier simulations, a don't care unit will generally maintain the value on the corresponding unit in the previous output. However, if the immediate predecessor is very dissimilar to the target, it is less likely to trigger assimilation. If the antepenultimate member of the sequence shows a strong similarity to the target it instead will be chosen as the trigger, and the penultimate member will be ignored. Furthermore, the similarity between the two potential triggers plays a role. If these two are identical, or markedly similar, the target will assimilate to the second of the two in all cases. The current simulations look at how these results explain the real language data.

In the data presented below, the expected (or *teacher*) output is given first, followed by the actual output of the network after 2000 iterations of training. The unspecified unit in the teacher is represented with an asterisk (*), and the corresponding output is given in boldface.

For the harmonic roots, both potential triggers have the same value for [back]. This value is straightforwardly maintained onto the don't care unit as a result of the smoothness constraint.

(12) iker - nek

i	0100	0.053	0.824	0.173	0.059
e	0010	0.036	0.149	0.853	0.032
e	*010	0.030	0.093	0.910	0.023

(13) kapu - nak

a	1010	0.951	0.099	0.911	0.086
u	1101	0.962	0.817	0.171	0.833
a	*010	0.947	0.120	0.886	0.111

In the mixed roots, the examples all contain a high front vowel in one syllable and a back vowel elsewhere. In these examples, the vectors representing both the suffix vowel and the back root vowel differ significantly from the root vowel *i*. Thus it is expected that even when *i* immediately precedes the underspecified vowel, it will exert little assimilatory influence. In addition, the first pattern in the string is strong similar to the target, and so is expected to have a strong influence. This is in fact the case, both in the simulation and in the real language data.

(14) taxi - nak

a	1 0 1 0	0.811	0.106	0.878	0.032
i	0 1 0 0	0.231	0.781	0.240	0.035
a	* 0 1 0	0.700	0.200	0.785	0.029

(15) bika - nak

i	0 1 0 0	0.134	0.877	0.130	0.050
a	1 0 1 0	0.845	0.110	0.883	0.025
a	* 0 1 0	0.870	0.071	0.930	0.029

However, the situation changes when the root contains a sequence of non-low front vowels. In these examples the target is preceded by a sequence of identical vectors. Here the identity of temporal context is the strongest factor, and the don't care unit is expected to assume the value of its immediate predecessor. Again, this behavior parallels the Hungarian facts.

(16) aspirin - nek

a	1 0 1 0	0.758	0.252	0.750	0.060
i	0 1 0 0	0.138	0.794	0.198	0.014
i	0 1 0 0	0.175	0.671	0.330	0.026
e	* 0 1 0	0.174	0.633	0.372	0.032

VI. Conclusion

To summarize, although the expected pattern in Hungarian is that all vowels of a word will agree in backness, certain front vowels in some environments respect this pattern, and in other environments do not. This more complex behavior is a function of both segmental identity and temporal context. Here I have suggested a processing treatment of the Hungarian facts which predicts harmonic behavior for these vowels on the basis of the overall similarity relationships among the vowels of the word.

A number of factors argue in favor of such an analysis. First, if offers a simpler and more explanatory account of Hungarian data. The vowels which exhibit transparent behavior, and the environments in which this behavior will change, must be stipulated arbitrarily under traditional accounts, while both follow automatically from the account proposed here. Second, this account makes strong claims about the existence of possible harmony patterns. As was demonstrated above, it is a general property of the sequential network that the assimilation process is sensitive to similarity in the temporal context. This predicts substantive constraints on what patterns of harmony may or may not exist. Whether these predictions can be maintained as a general principle requires further research, but available data suggest them to be correct.

Finally, although this accounts suggests that modifications of the autosegmental treatment of harmony are necessary, it is heavily influenced by the autosegmental notion of assimilation as the spread of a value for a particular phonetic feature. As the simulations demonstrate, a properly constrained treatment of temporal spread correctly predicts those harmonic patterns which exist in Hungarian, while failing to produce non-attested patterns. This result argues strongly in favor of the use of processing models as sources of constraint and explanation which can potentially enrich linguistic theory.

Notes

[1] I am grateful to Jeff Elman, Rob Kluender, Steve Poteet, Robert Port, George Lakoff, Sanford Schane, Gary Cottrell, Ann Thyme, Errapel Mejias-Bikandi and Kathleen Carey for useful comments and discussion.

[2] Results reported here are from a recurrent network with two input (plan) units, seven output and consequently seven state units, and six hidden units. The learning rate in simulations was 0.1; *mu* (the multiplier on the recurrent connections from each state unit to itself) was 0.6, and momentum was set to 0.

Bibliography

Archangeli, Diane. 1984. Underspecification in Yawelmani Phonology and Morphology. MIT PhD dissertation.

Archangeli, Diane and Douglas Pullyblank. 1986. The Content and Structure of Phonological Representations. ms.

Clements, George N. 1976. Vowel harmony in non-linear generative phonology. IULC.

Goldsmith, John. 1976. Autosegmental Phonology. PhD dissertation, MIT. Published 1979, New York: Garland Press.

Goldsmith, John. 1985. Vowel harmony in Khalkha Mongolian, Yaka, Finnish, and Hungarian. Phonology Yearbook 2. 251-275.

Goldsmith, John. 1989. Autosegmental and Metrical Phonology. Basil Blackwell.

Hulst, H. van der. 1985. Vowel harmony in Hungarian: A comparison of segmental and autosegmental analyses. In van der Hulst and Smith 1985.

Hulst, H. van der and Norval Smith (eds) 1985. Advances in nonlinear phonology. Dordrecht: Foris.

Jensen, J.T. 1978. A reply to "Theoretical implications of Hungarian vowel harmony". Linguistic Inquiry 9, 89-97.

Jordan, Michael. 1986. Serial Order: a Parallel Distributed Processing Approach. Institute for Cognitive Science Report #8604. UC San Diego.

Kontra, M. and Catherine Ringen. 1987. Stress and harmony in Hungarian loanwords. in Redei 1987.

McClelland, James L., Daved E. Rumelhart, and the PDP Research Group. 1986. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 2: Psychological and Biological Models. Cambridge, Mass.: MIT Press/Bradford Books.

Pullyblank, Douglas. 1988. Underspecification, the feature hierarchy, and Tiv vowels. Phonology S.2.

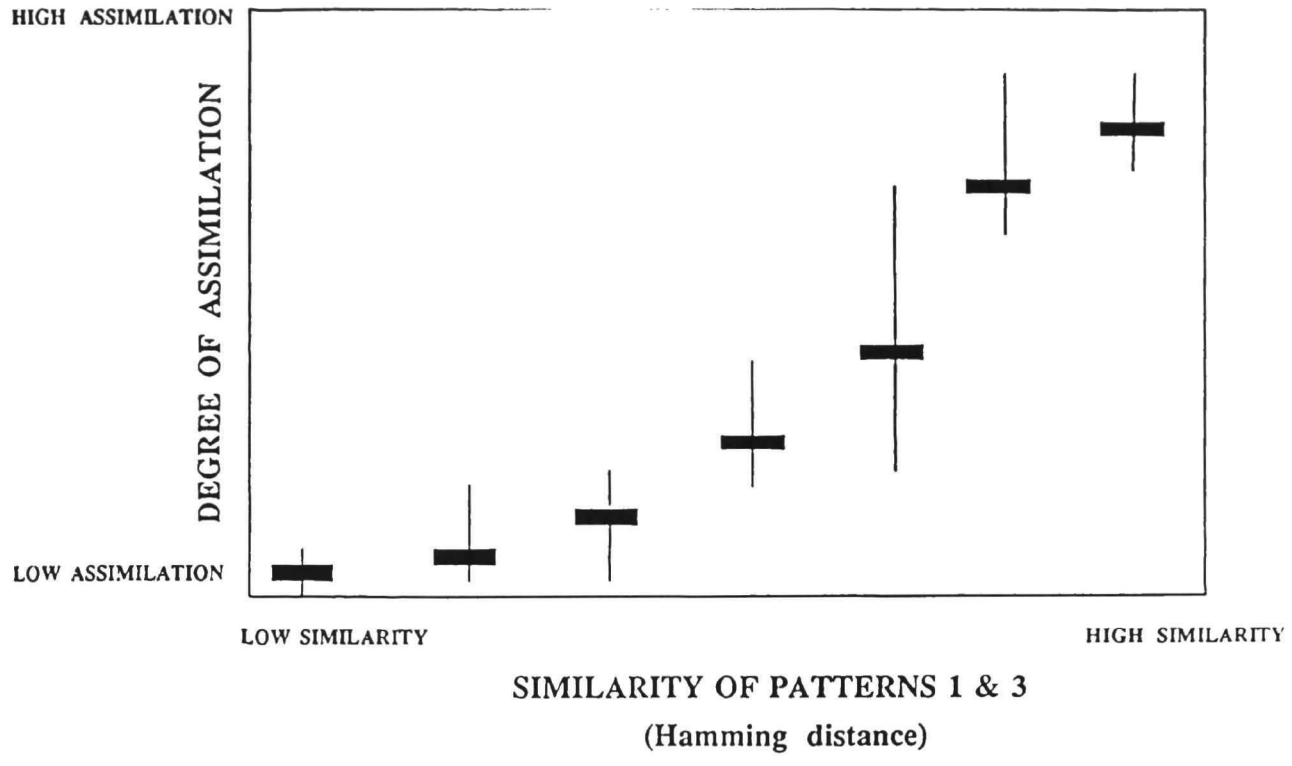
Redei, K. (ed) 1987. Studien sur Phonologie und Morphonologie der uralischen Sprachen. Studia Uralica, Wein: Verbund der wissenschaftlichen Gesellschaften Osterreichs. 81-96.

Ringen, Catherine. 1988. Transparency in Hungarian Vowel Harmony. Phonology Yearbook 5.2.

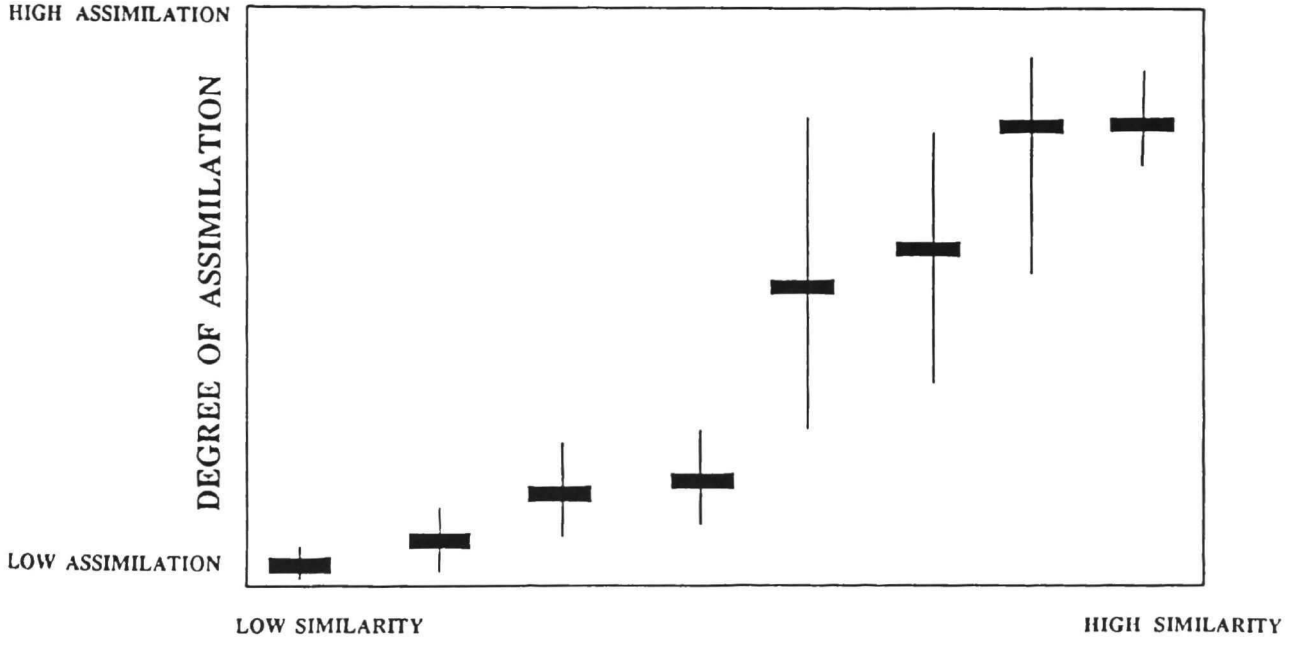
Rumelhart, David E., James L. McClelland, and the PDP Research Group. 1986. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1: Foundations. Cambridge, Mass.: MIT Press/Bradford Books.

Vago, Robert M. 1976. Theoretical implications of Hungarian vowel harmony. Linguistic Inquiry 7, 243-263.

Graph I



Graph II



Recency Preference and Garden-Path Effects

Edward Gibson
Department of Philosophy, Carnegie Mellon University
Pittsburgh, PA 15213-3890, USA
electronic mail: gibson@cs.cmu.edu

Abstract

Following Fodor (1983), it is assumed that the language processor is an automatic device that maintains only the best of the set of all compatible representations for an input string. One way to make this idea explicit is to assume the serial hypothesis: at most one representation for an input string is permitted at any time (*e.g.*, Frazier & Fodor (1978), Frazier (1979), and Pritchett (1988)). This paper assumes an alternative formulation of local memory restrictions within a parallel framework. First of all, it is assumed that there exists a number of structural properties, each of which is associated with a processing load. One structure is preferred over another if the processing load associated with the first structure is markedly lower than the processing load associated with the second. Thus a garden-path effect results if the unpreferred structure is necessary for a grammatical sentence.

This paper presents three structural properties within this framework: the first two– the Properties of Thematic Assignment and Reception– derivable from the θ-Criterion of Government and Binding Theory (Chomsky (1981)); and the third– the Property of Recency Preference– that prefers local attachments over more distant attachments. This paper shows how these properties interact to give appropriate predictions– garden-path effects or not– for a large array of local ambiguities.

1 Introduction

Perhaps the best-known theory of garden-path effects is that developed by Frazier and her colleagues (Frazier & Fodor (1978), Frazier (1979), Frazier & Rayner (1982)). This theory assumes the serial hypothesis: that at most one representation can be maintained for the input string at each parse state. In order to decide which structure to choose at a given parse state, the principles of Minimal Attachment and Late Closure are invoked. These principles are given in (1) and (2) respectively (from Frazier & Rayner (1982)):

(1) Minimal Attachment: Attach incoming material into the phrase-marker being constructed using the fewest nodes consistent with the well-formedness rules of the language.

(2) Late Closure: When possible, attach incoming lexical items into the clause or phrase currently being processed (*i.e.*, the lowest possible nonterminal node dominating the last item analyzed).

The principles of Minimal Attachment and Late Closure inside a serial processing model correctly predict a large array of garden-path effects and preferred readings of ambiguous input. Consider the following well-known examples:[1]

(3) # The horse raced past the barn fell.

(4) a. # Since she jogs a mile seems light work.

b. Bill thought John died yesterday.

Minimal attachment predicts the garden-path effect in (3). At the point of parsing the word *raced*, fewer nodes need to be constructed for the matrix verb reading than for the reduced relative clause reading, so the matrix verb reading is preferred. Since it is the reduced relative reading that is necessary for (3), a garden-path effect results.

Late Closure predicts the garden-path effect in (4a) and preferred reading in (4b). In (4a), there is a syntactic ambiguity at the point of parsing the noun phrase *a mile*: this NP may attach as either direct object of the verb *jogs* or as subject of the matrix clause to follow. Since the direct object attachment is within the same clause as the preceding words, this attachment is preferred by the principle of Late Closure. However, the matrix subject attachment is the attachment that is necessary for a successful parse of (4a). Thus a garden-path effect is correctly predicted.

Similarly, the nominal adverb *yesterday* can attach to either the embedded clause or to the matrix clause in (4b). Since the embedded clause occurs more recently in the input string, this attachment is preferred. Thus there is a preferred reading of (4b): that which associates *yesterday* with the embedded clause.

Although the principles of Minimal Attachment and Late Closure correctly account for numerous other garden-path effects, they also make a number of incorrect predictions, as noted in Pritchett (1988). For example, it is predicted that (5) should induce a garden-path effect:

(5) John knew Bill liked Sue.

---

[1] I will prefix sentences that are difficult to process with the symbol "#".

The NP *Bill* may attach in one of two locations: as direct object of the verb *knew* or as subject of the complement clause of the verb *knew*. Since the direct object attachment requires fewer nodes, it is preferred because of the principle of Minimal Attachment. Thus when the verb *liked* is processed, reanalysis must take place and a garden-path effect results. Since people have little difficulty parsing sentences like (5), Frazier's principles are in error in this case.

As a result, Pritchett proposes an alternative processing theory which collapses the Principles of Minimal Attachment and Late Closure into a single parsing principle:

(6) Σ-Attachment: Every principle of the Syntax attempts to be satisfied at every point during processing.

The syntactic principles that Pritchett assumes are those from Government and Binding Theory (Chomsky (1981), Chomsky (1986a)). In particular he appeals to the θ-Criterion:

(7) The θ-Criterion: Each argument bears one and only one θ-role (thematic role) and each θ-role is assigned to one and only one argument (Chomsky (1981) p. 36).

The garden-path effect in (3) is derived in Pritchett's theory as follows. When the word *raced* is input it can attach either as matrix verb or as a reduced relative clause modifier of *the horse*. In the matrix verb attachment, the NP *the horse* receives a thematic role from the verb *raced*. No such thematic role is assigned to *the horse* in the modifier attachment. Thus the matrix verb attachment is locally preferred since it better satisfies the θ-Criterion than does the modifier attachment. When the word *fell* is processed, no attachments are possible and reanalysis is necessary to obtain a parse for the sentence. Thus the garden-path status of (3) is predicted.

In order to account for the non-garden-path status of (5), Pritchett hypothesizes the existence of the Theta Reanalysis Constraint:

(8) Theta Reanalysis Constraint: Syntactic reanalysis which interprets a θ-marked constituent as outside its current θ-Domain is costly.

(9) θ-Domain: α is in the γ θ-Domain of β iff α receives the γ θ-role from β or α is dominated by a constituent that receives the γ θ-role from β.

Consider (5) with respect to the Theta Reanalysis Constraint. At the point of parsing the word *liked*, reanalysis is necessary. The NP *Bill* initially receives its θ-role from the verb *knew*; after reanalysis this NP receives its θ-role from the verb *liked*. Pritchett hypothesizes that this reanalysis is not costly because of the Theta Reanalysis Constraint: after reanalysis the θ-marked constituent *Bill* is still within its original θ-Domain since the θ-role initially assigned to *Bill* is now assigned to a constituent dominating *Bill*, the complementizer phrase *Bill liked*.[2,3]

Note that the definition of the Theta Reanalysis Constraint still predicts a garden-path effect in (3). At the parse state just before the word *fell* is input, the NP *the horse* receives the θ-role AGENT from the verb *raced*. When the word *fell* is input, the NP *the horse* must be reanalyzed as THEME of *fell* rather than as AGENT of *raced*. Since this reanalysis interprets the NP *the horse* as outside its original θ-Domain, this reanalysis is expensive and a garden-path effect results.

In addition, Pritchett's theory predicts a garden-path effect in (4a), thus partially collapsing Frazier's principles of Minimal Attachment and Late Closure. However, Pritchett's theory does not predict the preferred reading in (4b): no principle of the syntax is better satisfied by attaching the adverb *yesterday* to the embedded rather than to the matrix clause. Thus the preferred reading of (4b) is unexplained in Pritchett's theory: his theory will be forced to include a principle like Late Closure in order to account for examples like (4b).

Furthermore, both the theories of Pritchett and Frazier make incorrect predictions with respect to the data in (10):

(10) a. I gave her earrings on her birthday.

b. I gave her earrings to Sally.

Although the input string *I gave her earrings* is ambiguous between two possible readings, neither is difficult to process, as is demonstrated by the lack of garden-path effects in either of the sentences in (10). Since both Frazier's and Pritchett's models are serial models, there can be at most one representation for the input string *I gave her earrings*.

---

[2]The category complementizer phrase (CP) is the equivalent of the traditional *S'* node (Chomsky (1986b). Furthermore, tense and agreement information are assumed to reside in the category *Infl*. The category IP (Infl phrase) is the modern equivalent of the traditional *S* node.

[3]It turns out that there are difficulties with Pritchett's analysis when Case Theory is considered. If the Theta Reanalysis Constraint is not to be stipulative, then it should follow from (6). Since Pritchett appeals to Case Theory in many of his garden-path derivations, there should be a corresponding Case Reanalysis Constraint. However there cannot be such a constraint because of examples like (5). If the Case Filter is to be locally satisfied, then the preferred attachment of the NP *Bill* is as direct object of the verb *knew* so that it receives accusative Case. Case reanalysis is necessary when the verb *liked* is encountered: the noun *Bill* now receives nominative Case. Since this sentence is not difficult to process, this reanalysis should not be costly. Thus either (5) violates the Case Reanalysis Constraint or there is no Case Reanalysis Constraint and hence the Theta Reanalysis Constraint is stipulative.

Frazier's model therefore predicts a garden-path effect for one of the two sentences. Since Pritchett's model allows reanalysis as long as the Theta Reanalysis Constraint is not violated, it is necessary to check whether or not this constraint is violated by the parse of one of the sentences in (10). Consider the state of the parse after the input *I gave her earrings* has been processed. Since the verb *give* assigns two thematic roles, the representation that would best satisfy the θ-Criterion has both of these roles assigned. Thus Pritchett's theory predicts that the preferred reading at this parse state represents *her* and *earrings* as separate noun phrases, each receiving thematic roles from *gave*. Hence reanalysis is necessary in order to parse (10b). Furthermore, this reanalysis violates the Theta Reanalysis Constraint, since the NP *earrings* must be reanalyzed as within a new θ-Domain. Thus it is incorrectly predicted that (10b) should induce a garden-path effect.

I propose here that the difficulties encountered by these models can be overcome by the use of similar principles inside a constrained parallel model (*cf.* Gibson & Clark (1987), Clark & Gibson (1988)). By including a principle like Late Closure within a constrained parallel model, two structures can be maintained in parallel for the sentences in (10). Thus neither of these sentences will cause a garden-path effect.

2 A Parallel Model of Sentence Processing

Following Fodor (1983), it is assumed that the language processor is an automatic device that maintains only the best of the set of all compatible representations for an input string. One way to make this idea explicit is to assume the serial hypothesis: at most one representation for an input string is permitted at any time (*e.g.*, Frazier & Fodor (1978), Frazier (1979), and Pritchett (1988)). This paper assumes an alternative formulation of local memory restrictions within a parallel framework.[4] First of all, it is assumed that there exists a number of structural properties, each of which is associated with a processing load, in Processing Load Units or PLUs.[5] One structure is preferred over another if the processing load associated with the first structure is markedly lower than the processing load associated with the second.[6] That is, I assume there exists some preference quantity P corresponding to a processing load, such that if the processing loads associated with two representations for the same string differ by load P, then only the representation associated with the smaller of the two loads is pursued. Given the existence of a preference factor P, it is easy to account for garden-path effects and preferred readings of ambiguous sentences. Both effects occur because of a local ambiguity which is resolved in favor of one reading. In the case of a garden-path effect, the favored reading is not compatible with the whole sentence. Given two representations for the same input string that differ in processing load by at least the factor P, only the less computationally expensive structure will be pursued. If that structure is not compatible with the rest of the sentence and the discarded structure is part of a successful parse of the sentence, a garden-path effect results. If the parse is successful, but the discarded structure is compatible with another reading for the sentence, then only a preferred reading for the sentence has been calculated. Thus if we know where one reading of a (temporarily) ambiguous sentence becomes the strongly preferred reading, we can write an inequality associated with this preference:

(12)

$$\sum_{i=1}^{n} A_i x_i - \sum_{i=1}^{n} B_i x_i > P$$

where:

P is the preference factor in PLUs,
x_i is the number of PLUs associated with property i,
A_i is the number of times property i appears in the unpreferred structure,
B_i is the number of times property i appears in the preferred structure.

Three structural properties will be presented within this framework: the first two– the Properties of Thematic Assignment and Reception (Gibson (1990))– derivable from the θ-Criterion of Government and Binding Theory (Chomsky (1981)); and the third– the Property of Recency Preference– that prefers local attachments over more

---

[4] See Kurtzman (1985), Gorrell (1986) and Carlson & Tanenhaus (1989) for psycholinguistic evidence in favor of parallel processing.

[5] I also assume the existence of semantic and pragmatic properties that require processing load, but I will only consider structural properties here. The existence of further properties may help to explain the data presented in Crain & Steedman (1985) and Ni & Crain (1989).

[6] See Gibson (1990) for a description of how these same properties can be used to predict processing overload in sentences like (11):

(11) # The man that the woman that the dog bit likes eats fish.

distant attachments (*cf.* Frazier & Fodor (1978), Frazier (1979), Frazier & Rayner (1982)).

3 The Properties of Thematic Assignment and Reception

Recall the θ-Criterion:

(7) The θ-Criterion: Each argument bears one and only one θ-role (thematic role) and each θ-role is assigned to one and only one argument.

Note that the θ-Criterion can be violated in one of two ways: an argument can be missing a thematic role or a thematic role can be left unassigned. Thus I propose that the θ-Criterion has two corresponding parsing properties, each of which requires processing load:

(13) The Property of Thematic Reception (PTR):
Associate a load of x_{TR} PLUs of short term memory to each thematic element that is in a position that can receive a thematic role in some co-existing structure, but lacks a thematic role in the structure in question.

(14) The Property of Thematic Assignment (PTA):
Associate a load of x_{TA} PLUs of short term memory to each thematic role that is not assigned to a node containing a thematic element.

Note that the Properties of Thematic Assignment and Reception are stated in terms of *thematic* elements.[7] Thus the Property of Thematic Reception doesn't apply to functional categories, whether or not they are in positions that receive thematic roles. Similarly, if a thematic role is assigned to a functional category, the Property of Thematic Assignment does not notice until there is a thematic element inside this constituent.

Since the Properties of Thematic Assignment and Reception are both derived from the θ-Criterion, it is reasonable to assume as a default that the loads associated with these two properties is the same:

(15) $x_{TR} = x_{TA} = x_{\theta}$

Consider (5) with respect to the Properties of Thematic Assignment and Reception:

(5) John knew Bill liked Sue.

As pointed out earlier, the verb *knew* is ambiguous: either taking an NP complement or a CP complement. Thus the NP *Bill* may attach as either the direct object of the verb *knew* or as subject of the CP to come.[8]:

(16) a. $[_{IP}\ [_{NP}$ John] $[_{VP}$ knew $[_{NP}$ Bill]]]

b. $[_{IP}\ [_{NP}$ John] $[_{VP}$ knew $[_{CP}\ [_{IP}\ [_{NP}$ Bill] e]]]]

In (16a) the NP *Bill* is attached as the NP complement of *knew*. In this representation there is no load associated with either of the Properties of Thematic Assignment or Reception since no thematic elements need thematic roles and no thematic roles are left unassigned. In (16b) the NP *Bill* is the specifier of a hypothesized IP node which is attached as the complement of the other reading of *knew*. This representation is associated with at least x_{θ} PLUs since the NP *Mary* is in a position that can be associated with a thematic role (the subject position), but does not yet receive one in this structure. No load is associated with the Property of Thematic Assignment, however, since both thematic roles of the verb *knew* are assigned to nodes that contain thematic elements.

Since there is no difficulty in processing sentence (5), the load difference between the structures in (16) cannot be greater than P PLUs, the preference factor assumed in inequality (12). Thus the inequality in (17) is obtained:

(17) $x_{\theta} \leq P$

Since the load difference between the two structures is not sufficient to cause a strong preference, both structures are maintained. Note that this is a crucial difference between the theory presented here and the theory presented in Frazier & Fodor (1978), Frazier (1979) and Pritchett (1988). In each of these theories, only one representation can be maintained, so that either (16a) or (16b) would be preferred at this point. As noted earlier, Pritchett's theory accounts for the lack of difficulty experienced in parsing (5) by appeal to the Theta Reanalysis Constraint, (8). However, no such stipulation is required in the theory presented here: all reanalysis is assumed to be expensive.

---

[7] Following early work in linguistic theory, two kinds of categories are distinguished: *functional* categories and *thematic* or *content* categories (see, for example, Fukui & Speas (1986) and Abney (1987) and the references cited in each). Thematic categories include nouns, verbs, adjectives and some prepositions; functional categories include determiners, complementizers, and inflection markers. There are a number of properties that distinguish functional elements from thematic elements, the most crucial being that functional elements mark grammatical or relational features while thematic elements pick out a class of objects or events.

[8] I assume some form of hypothesis-driven node projection so that noun phrases are projected to the categories that they specify (Gibson (1989)).

Now consider once again (3):

(3) # The horse raced past the barn fell.

The structure for the input *the horse raced* is ambiguous between at least the two structures in (18):

(18) a. $[_{IP}\ [_{NP}$ the horse $]\ [_{VP}$ raced $]]$

b. $[_{IP}\ [_{NP}$ the $[_{N'}\ [_{N'}$ horse$_i$ $]\ [_{CP}\ O_i$ raced $]\ ]]\ [_{I'}\ ]]$

Structure (18a) has no load associated with it due to either the PTA or the PTR. Crucially note that the verb *raced* has an intransitive reading so that no load is required via the Property of Thematic Assignment. On the other hand, structure (18b) requires a load of $2 * x_\theta$ PLUs since 1) the noun phrase *the horse* is in a position that can receive a thematic role, but currently does not and 2) the operator O_i is in a position that may be associated with a thematic role, but is not yet associated with one.[9] Thus the difference between the processing loads of structures (18a) and (18b) is $2 * x_\theta$ PLUs. Since this sentence is a strong garden-path sentence, it is hypothesized that a load difference of $2 * x_\theta$ PLUs is greater than the allowable limit, P PLUs:

(19) $2 * x_\theta > P$

Consider now (20), a sentence whose structure and local ambiguities are very similar to those in (3):

(20) The bird found in the room was dead.

Although the structures and local ambiguities in (20) and (3) are similar, (3) causes a garden-path effect while, surprisingly, (20) does not. To determine why (20) is not a garden-path sentence we need to examine the local ambiguity when the word *found* is read:

(21) a. $[_{IP}\ [_{NP}$ the bird $]\ [_{VP}\ [_{V'}\ [_{V}$ found $]]]]$

b. $[_{IP}\ [_{NP}$ the $[_{N'}\ [_{N'}$ bird$_i$ $]\ [_{CP}\ O_i$ found $]\ ]]\ [_{I'}\ ]]$

The crucial difference between the verb *found* and the verb *raced* is that *found* is obligatorily transitive, while *raced* is optionally intransitive. Since the θ-grid of the verb *found* is not filled in structure (21a), this representation is associated with x_θ PLUs of memory load. Like structure (18b), structure (21b) requires $2 * x_\theta$ PLUs. Thus the difference between the processing loads of structures (21a) and (21b) is $2 * x_\theta - x_\theta$ PLUs = x_θ PLUs. Since this difference is less than the preference factor P, both structures are maintained and a garden-path effect is avoided as desired.[10]

The Properties of Thematic Assignment and Reception make many further garden-path predictions, among them:

(23) a. # I put the candy on the table in my mouth.

b. # The Russian women loved died.

c. # John told the man that Mary kissed that Bill saw Phil.

See Gibson (1990) for derivations of the garden-path status of these sentences within the framework described here.

4 The Property of Recency Preference

Although the Properties of Thematic Assignment and Reception account for numerous garden-path effects and preferred readings of ambiguous input, they do not account for the Late Closure effects presented in (4):

(4a) # Since she jogs a mile seems light work.

---

[9] In fact, this operator will be associated with a thematic role as soon as a gap-positing algorithm links it with the object of the passive participle *raced*. However, when the attachment is initially made, no such link yet exists: the operator will initially be unassociated with a thematic role.

[10] Note that the principle of Minimal Attachment makes the wrong prediction in (20): this example is wrongly predicted to be identical to (3) and thus a garden-path.

Pritchett attempts to account for the lack of garden-path effect in (20) by altering the Theta Reanalysis Constraint in such a way that (20) does not violate the new constraint while garden-path sentences like (3) still do. Crucial to his argument is his claim that there is a strong preference for the matrix clause reading of the word *found* when it is initially input. However, Kurtzman (1985) found that both matrix clause and reduced relative clause readings of verbs like *found* were equally easy to process as soon as they were read, contrary to Pritchett's claim. For example, consider the sentences in (22):

(22) a. The monkeys chased his...

b. The monkeys chased were...

Kurtzman found that partial sentences like those in (22) did not vary significantly in parsing difficulty. Thus Pritchett's claim that words like *found* or *chased* are initially analyzed as matrix verbs appears to be incorrect. Furthermore, note that a parallel model of processing makes exactly the right predictions here.

(4b) Bill thought John died yesterday.

First consider (4a). There are two possible attachments of the NP *a mile*: 1) as direct object of the verb *jogs*; and 2) as subject of the matrix clause to follow. Consider these two structures:

(24) a. [$_{IP}$ [$_{CP}$ since [$_{IP}$ [$_{NP}$ she] [$_{VP}$ [$_{V'}$ [$_{V}$ jogs] [$_{NP}$ a mile]]]]] [$_{IP}$]]

b. [$_{IP}$ [$_{CP}$ since [$_{IP}$ [$_{NP}$ she] [$_{VP}$ jogs]]] [$_{IP}$ [$_{NP}$ a mile] [$_{I'}$]]]

There is no load associated with structure (24a) by the Properties of Thematic Assignment and Reception since all arguments receive thematic roles and all thematic roles are assigned. Structure (24b), on the other hand is associated with x_θ PLUs since the NP *a mile* currently lacks a thematic role in this structure. However this is the only load associated with (24b). Thus the load difference between the two structures is only x_θ PLUs by the Properties of Thematic Assignment and Reception, not enough to cause a garden-path effect.

It turns out that this problem is solved by incorporating a structural property derived from a principle similar to Late Closure into the framework described thus far. The intuition behind the Principle of Late Closure and its predecessors (Kimball (1973)) is that new structures prefer to be attached to structures associated with more recent words in the input string. For example, in (4b), the preferred attachment for the adverb *yesterday* is as modifier of the more recently occurring clause: the most deeply embedded one. The property that I present here, the Property of Recency Preference (PRP), makes this intuition explicit.

(25) The Property of Recency Preference (PRP):
The load associated with the structure resulting from an attachment of structure X = (number of more recent words that would also allow an attachment of structure X) $*x_{RP}$ PLUs.

Hence given two possible attachment sites, the structure resulting from attachment to the more recent word will be associated with no load via the Property of Recency Preference, while the structure resulting from attachment to the less recent word will be associated with x_{RP} PLUs.

Consider the PRP with respect to sentence (4b). At the point of parsing the word *yesterday*, there are two possible attachments: either as a modifier of the embedded clause headed by *died* or as modifier of the matrix clause headed by *thought*. The structure that results from attaching this adverb as modifier of the embedded clause is associated with no load via the Property of Recency Preference, since the word *died* is the most recently occurring attachment site in the input string. However, the structure that results from attaching this adverb to the matrix clause is associated with x_{RP} PLUs, since there is a more recent word in the input string, the word *died*, to which this adverb could attach. Since the interpretation which links the adverb *yesterday* with the embedded clause is the strongly preferred reading in (4b), I hypothesize that the load difference between the structures resulting from the two possible attachment sites is significant:

(26) $x_{RP} > P$

The existence of a garden-path effect in (4a) can now be explained. Structure (24a) is associated with no load via the Property of Recency Preference since the NP *a mile* is attached to a structure headed by the most recent word in the input string, *jogs*. Structure (24b), on the other hand, represents attachment to the CP headed by *since*. Thus this structure is associated with x_{RP} PLUs by the PRP. Since this structure also contains a thematic role-less NP, the total load associated with this structure is $x_{RP} + x_\theta$ PLUs. Structure (24a) is associated with no load whatsoever. Thus the load difference between the two structures is $x_{RP} + x_\theta$ PLUs, enough to cause a strong local preference and hence a garden-path.

Although the Property of Recency Preference is very similar to Frazier's principle of Late Closure, there is an important difference. Unlike the principle of Late Closure, the PRP is not stipulated to act on its own: it can interact with other properties. Thus we expect to find situations where the PRP interacts with the Properties of Thematic Assignment and Reception in the prediction of behavior in ambiguous situations. In fact, many such cases occur. First consider (27), a sentence whose garden-path effect is unexplained without the PRP:

(27) # I convinced her children are noisy.

(27) is locally ambiguous over the span of the words *her children*. Two parses for the input string *I convinced her children* are given in (28):

(28) a. [$_{IP}$ [$_{NP}$ I] [$_{VP}$ [$_{V'}$ [$_{V}$ convinced] [$_{NP}$ her children]]]]

b. [$_{IP}$ [$_{NP}$ I] [$_{VP}$ [$_{V'}$ [$_{V}$ convinced] [$_{NP}$ her] [$_{CP}$ [$_{IP}$ [$_{NP}$ children]]]]]]

In structure (28a) the words *her children* correspond to the noun phrase object of *convinced*. In (28b), the word *her* is the noun phrase object of *convinced* and the word *children* is the subject of the complement clause of *convinced*. Structure (28a) is strongly preferred over (28b) and a garden-path effect results for (27).

In order to see how this garden-path effect can be predicted in this framework, consider the loads associated with each of the structures in (28). First, let us consider the Property of Recency Preference. The attachment of the NP representation of the word *children* to form (28a) requires no PRP load since the attachment involves the most recent word in the input string, *her*. On the other hand, the attachment of *children* as the subject of the CP complement to *convinced* involves making an attachment to the verb *convinced*, a less recent word in the input string. Thus there is a load of x_{RP} PLUs associated with (28b).

Consider now the Properties of Thematic Assignment and Reception. Structure (28a) has a load of x_θ PLUs since the verb *convinced* has a yet unassigned thematic role. Structure (28b) is associated with a load of x_θ PLUs since the noun phrase *children* requires a thematic role and does not yet receive one. Thus the total load difference between the two structures is $x_\theta + x_{RP} - x_\theta$ PLUs = x_{RP} PLUs. Thus (27) induces a garden-path effect, as desired.

Furthermore, the combination of the properties also explains the lack of difficulty in the sentences in (10):

(10a) I gave her earrings on her birthday.
(10b) I gave her earrings to Sally.

In order to formulate an inequality representing the state of affairs in (10), consider the parse state after the word *earrings* is read:

(29) a. [$_{IP}$ [$_{NP}$ I] [$_{VP}$ [$_{V'}$ [$_{V}$ gave] [$_{NP}$ her earrings]]]]

b. [$_{IP}$ [$_{NP}$ I] [$_{VP}$ [$_{V'}$ [$_{V}$ gave] [$_{NP}$ her] [$_{NP}$ earrings]]]]

The verb *gave* subcategorizes for either two noun phrases or for a noun phrase and a prepositional phrase (*cf.* Kayne (1984), Larson (1988)). First consider the load associated with the structures in (29) with respect to the Property of Recency Preference. The derivation of this load is similar to that for the structures in (28). Structure (29a) is associated with no load by the PRP since *earrings* attaches to *her*, the most recent word. However, structure (29b) is associated with a load of x_{RP} PLUs since the attachment necessary to form this structure involves a less recent word, *gave*. Now consider the loads associated with the structures in (29) with respect to the Properties of Thematic Assignment and Reception. Structure (29a) is associated with a load of x_θ PLUs since a thematic role is yet unassigned by the verb *gave* in this structure. On the other hand, structure (29b) is associated with no load due either the PTA or the PTR since all thematic roles are assigned and all thematic elements receive thematic roles. Note that this is the crucial difference between (10) and (27). Thus the load difference between structures (29a) and (29b) is $x_{RP} - x_\theta$ PLUs. Since neither (10a) nor (10b) is a garden-path sentence, I hypothesize that this load difference is not significant:

(30) $x_{RP} - x_\theta \leq P$

Thus we have the following inequalities:

(31) a. $x_\theta \leq P$

b. $2 * x_\theta > P$

c. $x_{RP} > P$

d. $x_{RP} - x_\theta \leq P$

This set of inequalities is consistent. It identifies the solution space depicted in Figure 1.

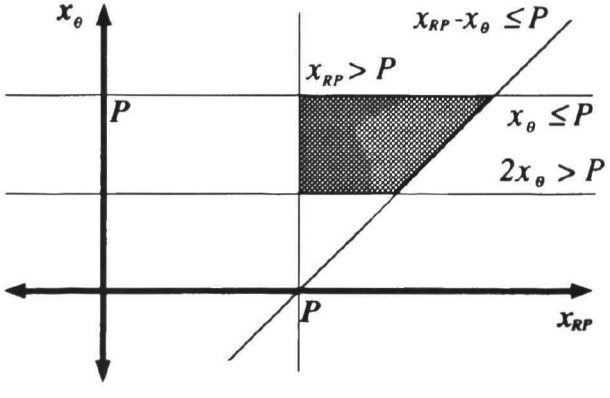


Figure 1: The Solution Space for the Inequalities in (31)

5 Conclusions

This paper has presented a parallel model of human sentence processing in which it is assumed that the ease or difficulty in parsing a given sentence is due to multiple coexisting properties of the sentence. Three independent properties were presented: two that follow from the θ-Criterion, and the third, a property that prefers attachments involving more recent words. These three properties were shown to interact to correctly predict previously unexplained results with respect to a number of local ambiguities. Furthermore, no stipulations regarding reanalysis are necessary: all reanalysis is assumed to be expensive. Thus the framework presented here makes better predictions with respect to the data considered here than the currently accepted serial models of language processing.

6 References

Abney (1987) *The English Noun Phrase in its Sentential Aspect*, MIT Ph.D. dissertation, Cambridge, MA.

Carlson, G.N., & Tanenhaus, M.K. (1989) "Thematic Roles and Language Comprehension", in Wilkins, W. (ed.), *Syntax and Semantics Volume 21: Thematic Relations*, Academic Press, San Diego, CA.

Chomsky, N. (1981) *Lectures on Government and Binding*, Foris, Dordrecht, The Netherlands.

Chomsky, N. (1986a) *Knowledge of Language: Its Nature, Origin and Use*, Praeger Publishers, New York, NY.

Chomsky, N. (1986b) *Barriers*, Linguistic Inquiry Monograph 13, MIT Press, Cambridge, MA.

Clark, R. & Gibson, E. (1988) "A Parallel Model for Adult Sentence Processing", *Proceedings of the Tenth Cognitive Science Conference*, McGill University, Montreal, Quebec.

Crain, S., & Steedman, M. (1985) "On Not Being Led Up the Garden Path: the Use of Context by the Psychological Parser", in D. Dowty, L. Karttunen, & A. Zwicky (eds.), *Natural Language Processing: Psychological, Computational and Theoretical Perspectives*, Cambridge University Press, Cambridge, U.K.

Fodor, J.A. (1983) *Modularity of Mind*, MIT Press, Cambridge, MA.

Frazier, L. (1979) *On Comprehending Sentences: Syntactic Parsing Strategies*, University of Massachusetts Ph.D. dissertation.

Frazier, L., & Fodor, J.D. (1978) "The Sausage Machine: A New Two-stage Parsing Model", *Cognition* 6, pp. 291-325.

Frazier, L. & Rayner, K. (1982) "Making and Correcting Errors During Sentence Comprehension: Eye Movements in the Analysis of Structurally Ambiguous Sentences", *Cognitive Psychology* 14, pp. 178-210.

Fukui, N., & Speas, M. (1986) "Specifiers and Projections", *MIT Working Papers in Linguistics* 8, Cambridge, MA.

Gibson, E., & Clark, R. (1987) "Positing Gaps in a Parallel Parser," *Proceedings of the Eighteenth North East Linguistic Society Conference*, University of Toronto, Toronto, Ontario.

Gibson, E. (1989) "Parsing with Principles: Predicting a Phrasal Node Before Its Head Appears", *Proceedings of the First International Workshop on Parsing Technologies*, Carnegie Mellon University, 1989.

Gibson, E. (1990) "Memory Capacity and Sentence Processing", *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, Pittsburgh, PA.

Gorrell, P.G. (1987) *Studies of Human Syntactic Processing: Ranked-Parallel versus Serial Models*, University of Connecticut Ph.D. dissertation.

Kayne, R. S. (1984) *Connectedness and Binary Branching*, Foris, Dordrecht, The Netherlands.

Kimball, J. (1973) "Seven Principles of Surface Structure Parsing in Natural Language," *Cognition* 2, pp. 15-47.

Kurtzman, H. (1985) *Studies in Syntactic Ambiguity Resolution*, MIT Ph.D. dissertation.

Larson, R.K. (1988) "On the Double Object Construction," *Linguistic Inquiry*, 19, pp. 335-391.

Ni, W. and Crain, C. (1989) "How to Resolve Structural Ambiguities," *Proceedings of the Twentieth North East Linguistic Society Conference*, Pittsburgh, PA.

Pritchett, B. (1988) "Garden Path Phenomena and the Grammatical Basis of Language Processing", *Language* 64 pp. 539-576.

A Connectionist Treatment of Grammar for Generation

Nigel Ward

Computer Science Division
University of California at Berkeley

Abstract

Connectionist language generation promises better interaction between syntactic and lexical considerations and thus improved output quality. To realize this requires a connectionist treatment of grammar. This paper explains one way to do so. The basic idea is that constructions and their constituents are nodes in the same network that encodes world knowledge and lexical knowledge. The principal novelty is reliance on emergent properties. This makes it unnecessary to maké explicit syntactic choice or to build up representations of sentence structure. The scheme includes novel ways of handling constituency, word order and optional constituents; and a simple way to avoid the problems of instantiation and binding. Despite the novel approach, the syntactic knowledge used is expressed in a form similar to that often used in linguistics; this representation straightforwardly defines parts of the knowlege network. These ideas have been implemented in FIG, a 'flexible incremental generator.'

1 Introduction

A generator is faced with a great number of interdependent options: lexical, syntactic, and conceptual. To produce a good utterance a generator must arrive at a consistent set of choices. The best way to do this seems to be with explicit parallel consideration of all possible options and parallel computation of their interdependencies [Ward 89c]. This ensures that the relevant information is all available, something which is difficult for generators which make choices in sequence [Ward 89b].

For syntax, this implies considering many possible constructions at once. This technique appears to be used by human speakers – analysis of speech errors suggests that even normal speech is the result of competing 'plans' [Baars 80]. [Stemberger 85] realized that, in particular, human speakers can be modeled as having many 'phrase structure units' being 'partially activated' simultaneously, and stressed the importance of emergents.

I have written an intrinsically parallel generator, 'FIG,' motivated by considerations of cognitive modeling [Ward 89c]. FIG is a structured (aka localist) connectionist system. Structured connectionism allows parallelism while making it relatively easy to build, explain, and debug the system. (A distributed connectionist system would have other advantages but would be harder to develop.)

This paper presents a new approach to grammar for language generation. The key innovation is reliance on emergent properties, instead of making explicit choices and doing explicit structure-building. Despite the novel approach to processing, there is a declarative representation for linguistic knowledge.

To see that this approach works for syntactically non-trivial examples, consider that FIG's outputs include: "once upon a time there lived an old man and an old woman," "one day the old man went into the hills to gather wood," "a big peach bobbed down towards an old woman from upstream," "an old woman gave a peach to an old man," and "Mary was killed;" and corresponding Japanese sentences: "mukashi mukashi aru tokoro ni ojiisan to obaasan ga sunde imashita," "aru hi ojiisan wa yama e shibakari ni ikimashita," "kawakami kara ookii momo ga donburiko donburako to obaasan e nagarete kimashita," "ojiisan wa meeri ni momo o agemashita," and "meeri o koroshimashita."

Section 2 overviews contrasting and related treatments of syntax in generation, Section 3 summarizes the FIG approach to generation, Section 4 presents a representation for grammatical knowledge, Section 5 describes how this knowledge is used, Section 6 discusses the implementation, and Section 7 suggests fu-

[0]Thanks to Daniel Jurafsky, Robert Wilensky, and Dekai Wu. This research was sponsored by the Defense Advanced Research Projects Agency (DoD), monitored by the Space and Naval Warfare Systems Command under N00039-88-C-0292, and the Office of Naval Research under contract N00014-89-J-3205.

ture research directions.

2 Previous Research

Problems of grammar for generation have received a fair amount of attention in AI. Of the work which is concerned with the details of language, almost all is based on syntactic mechanisms adopted from linguistic theories. Most linguistic grammars come complete with mechanisms: transformations, parse-tree traversals, unification, and so on. Yet these mechanisms are not intended to be computational models. They are typically inspired by the goal of explaining sentence structure, a goal probably originally due to linguistics' focus on grammaticality.

Independent of linguistic theories, the most common metaphor for generation is as a sequence of choices among alternatives. For example, a generator may chose among words for a concept, among ways to syntactically realize a constituent, and among concepts to bind to a slot. These decisions are generally made in a fairly fixed (and generally top-down) order, thus most generators are not easily parallelizable.

In FIG grammar is important but grammaticality is not. Grammar is a tool used in the process of expressing meaning, not a goal in itself. Structure building, structure mapping, and explicit syntactic choice are dispensed with. In FIG the structure of the output is emergent, and choices are also largely emergent. One advantage of relying on emergents is that there is no need to order choices [Ward 89b], and thus the process is naturally parallelizable.

Previous connectionist research has in general not strayed far from traditional approaches to grammar. [Stolcke 89] directly implemented unification grammar. [Kalita and Shastri 87] implemented a standard symbolic generator [McDonald 1983]. Perhaps the most original connectionist generator is Gasser's CHIE [Gasser 88]. Yet even in CHIE there are choices (represented by neuron firings) and these happen sequentially, in order. The exact timing of firings seems crucial. CHIE freely uses winner-take-all subnetworks, which also cuts down on the amount of effective parallelism.

3 The FIG Approach to Generation

The task is generation of natural language from thoughts. (Most generators, presume that the input has been pre-processed by a 'what-to-say' component, and thus only need to take the 'message' and do some language-specific processing. I think this division of the task is unnecessary and unwise [Ward 88b].) Reduced to bare essentials, a generator's task is to get from concepts (what the speaker wants to express) to words (what he can say). From this point of view, the key problem in generation is computing the relevance (pertinence) of a particular word, given the concepts to express. Syntactic and other knowledge mediates this computation of relevance. Therefore FIG is based on word choice – every other consideration is analyzed in terms of how it affects word choice.

Processing in FIG is done with spreading activation in a network. The basic FIG algorithm is:

1. each node of the input conceptualization is a source of activation
2. activation flows through the network
3. when the network settles, the most highly activated word is selected and emitted
4. activation levels are updated to represent the new current state
5. steps 2 through 4 repeat until all of the input has been conveyed

An utterance is simply the result of successive word choices, thus FIG is incremental in a strong sense.

Activation represents relevance; flow of activation represents implications of relevance; and updating the activation of a node represents computing its relevance. The general direction of activation flow is from concepts to words, but nodes representing syntax 'redistribute' activation also, and feedback is pervasive.

The network must be designed so that, when it settles, the node which is most highly activated corresponds to the best next word. This paper discusses the design of the network structures which encode syntactic knowledge.

4 Knowledge of Syntax

FIG's treatment of syntax is based on Construction Grammar [Fillmore et al 89]. This approach describes the grammar of a language 'directly, in terms of a collection of grammatical constructions' [Fillmore 88]. The key characteristics here are that Construction Grammar is declarative and that the units of syntactic knowledge, namely constructions, are densely related to words, meaning, and each other.

The encodings of syntactic knowledge shown below are taken directly from FIG. These are for illustrative purposes only. I do not claim that these represent the facts of English, nor the best way to describe them in a grammar. The examples are intended simply to illustrate the representational tools and computational mechanisms available in FIG.

Figure 1 shows FIG's definition of **noun-phr**, representing a simplification of the English noun-phrase construction. This construction has three constituents:

```
(defp noun-phr
  (constituents (np-1  obl  article  ((article .8) (a-w .2) ))
                (np-2  opt  adjective((adjective .6)))
                (np-3  obl  noun     ((noun .9))) ))
```

Figure 1: The English Noun-Phrase Construction

```
(defp go-p
  (constituents (gp-1  obl go-w   ((go-w .1)))
                (gp-2  opt epart ((vparticle .6) (directionr .2)))
                (gp-3  opt noun  ((prep-phr .6) (destr .2)))
                (gp-4  opt verb  ((purpose-clause .6) (purposer .2))) ))
```

Figure 2: Representation of the Valence of "Go"

```
(defp ex-there
  (inhibit subj-pred passive)
  (constituents (et-1 obl therew ((therew .9)))
                (et-2 obl verb   ((verb .9)))
                (et-3 obl noun   ((noun .5))) ))
```

Figure 3: Representation of the Existential "There" Construction

```
(defw peachw    (cat noun)   (expresses momoc)  (grapheme "peach")
     (nebors (initial-phoner consnt-initial .5)) )

(defs noun   (maximals (noun-phr .4)))

(defw go-w   (cat verb)  (expresses ikuc)  (valence (go-p .2))
     (grapheme (inf "go") (past "went") (pastp "gone") (presp "going")) )

(defc introductoryc  (english (ex-there .1) (a-w .1)) )

(defr purposer (english (to2w .4) (purpose-clause .1))
     (japanese (ni-w .6)))
```

Figure 4: Some Knowledge Related to Constructions

np-1, **np-2**, and **np-3**. **np-1** and **np-3** are obligatory, **np-2** is optional. Glossing over the details for the moment, the list at the end of the definitions specifies how to realize the constituent. For example, **np-1** should be realized as an **article**, with the default being **a-w** (representing the word "a"), and so on.

Figure 2 shows the construction for the case frame of the word "go." First comes **go-w**, for the word "go," which is obligatory. Next come (optionally): a verb-particle representing direction (as in "go *away*" or "go *back* home" or "go *down* to the lake"), a prepositional phrase to express the destination, and a purpose clause.

Figure 3 shows the representation of the existential "there" construction, as in "there was a poor cobbler." The 'inhibit' field indicates that this construction is incompatible with the **passive** construction and also with **subj-pred**, the construction responsible for the basic SVO ordering of English.

Figure 4 shows knowledge about when and where constructions are relevant. Constructions are associated with words. For example **noun-phr** is the 'maximal' of **noun** (actually, of all nouns), and **go-p** is the 'valence' (case frame) of **go-w**. These links encode knowledge about constructions and their heads; other relations between words and constructions are discussed in Section 5.5. Constructions are also associated with the meanings they can express. For example, **ex-there** is listed under the concept **introductory**, representing that this construction is appropriate for introducing some character into the story, and **purpose-clause** is listed as a way to express the **purposer** relation. Constructions are also associated with other constructions. For example, the fourth constituent of **go-p** subcategorizes for **purpose-clause** (Figure 2); and there are associations among incompatible constructions, for example the 'inhibit' link between **ex-there** and **subj-pred** (Figure 3).

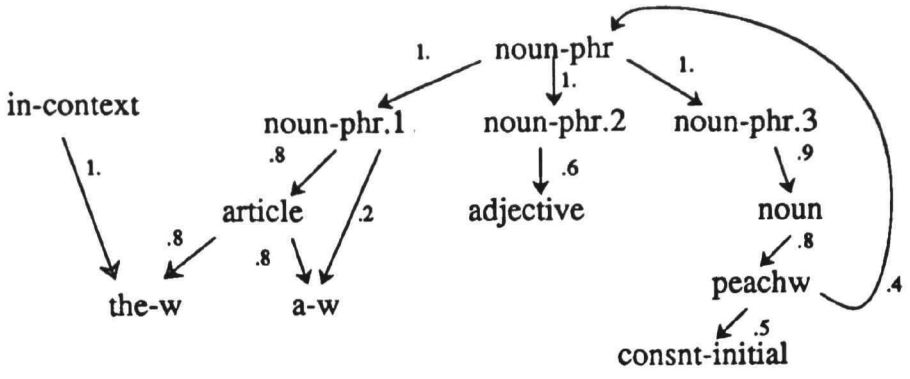


Figure 5: A Fragment of the Network

5 Using This Knowledge

While the above representations resemble those used by several modern linguistic theories, FIG uses them in a novel way.

5.1 Constructions in the Network

In FIG constructions and constituents are nodes in the knowledge network. Their activation levels represent their current relevance. Constructions receive activation from nodes linked to them, and transmit activation to other nodes. Figure 5 shows a fragment of FIG's network, where the numbers on the links are their weights. This is partially specified by the knowledge shown in the previous figures [1]. It is generally possible to give procedural interpretations to links. For example, the link from **peachw** to **noun-phr** 'advises' that: if you want to say a noun, you should 'consult' the knowledge in **noun-phr**.

Activation flow among the various nodes in the network provides, among other things, pervasive interaction among lexical choices and syntactic considerations. The rest of this section explains how this handles some basic functions of syntax.

5.2 Constituency

The links defined above suffice to handle constituency. Consider for example the fact that nouns need to be preceded by articles in this particular kind of noun phrase. Suppose that **peachw** is activated, perhaps because a **peachc** concept is in the input. Activation flows from **peachw** to **noun-phr**, from **noun-phr** to **article**, and from there to **a-w** and **the-w**. **a-w** also receives activation via a direct link from **noun-phr**.

In this way the relevance of a noun increases the relevance rating of articles. Provided that other activation levels are appropriate, this will cause some article to become the most highly activated word, and thus be selected and emitted. Note that FIG does not first choose to say a noun, then an article; both 'decisions' are considered and made together, as activation levels settle.

If there are additional sources of activation (such as **in-context**) then **the-w** will receive more activation, and thus "the" will be output. Also, if the node **vowel-initial** is more highly activated than the node **consnt-initial**, "an" instead of "a" will be output, thanks to the inflection mechanism (not described further).

There is also the question of specifying where a given concept should appear and what syntactic form it should take. This problem, subcategorization, is handled in FIG by simultaneously activating a concept and the syntactic form it should take. For example, the third constituent of **go-p** specifies that 'the direction of the going' be expressed as a 'verbal particle.' Activation will thus flow to an appropriate word node, such as **downw**, both via the concept filling the **directionr** slot and via the syntactic category **vparticle**. Thanks to this sort of activation flow, FIG tends to select and emit an appropriate word in an appropriate form [Ward 88a].

Syntactic considerations manifest themselves only through their effects on the activation levels of words. Syntax is never 'in control' of word choice [2]; the syntactic structure of the result is emergent.

[1] The mapping from s-expressions to network structures is not always trivial. For example, the link from **noun** to **peachw** comes from the statement that **peachw** had 'cat' **noun**; and the link from **peachw** to **noun-phr** is inherited by **peachw** from the 'maximals' information on **noun**.

[2] Post hoc examination of FIG output might make one think, for example, 'this exhibits the choice of the existential-there construction.' In FIG there is indeed an inhibit link between the nodes **ex-there** and **subj-pred**, and so when generating the network tends to reach a state where only one of these nodes is highly activated. The most highly activated construction can have a strong effect on word choices, which is why the appearance of syntactic choice arises.

5.3 Word Order

FIG is an incremental generator, that is, it selects and emits words one by one in order. At each time the activation level of a word must represent its *current* relevance. To this end, one job of constructions is to activate things which are currently syntactically appropriate. In FIG the current syntactic state is represented in the state of each construction node, namely, their activation levels and 'cursors.' The cursor of a construction points to the currently appropriate constituent and ensures that it is relatively highly activated. To be specific, the cursor gives the location of a 'mask' specifying the weights of the links from the construction to constituents. The mask specifies a weight of 1.0 for the constituent under the cursor, and for subsequent constituents a weight proportional to their closeness to the cursor [3]. This is parallelism among constituents.

For example, if the cursor of **noun-phr** points to **np-1**, then articles will receive a large proportion of the activation of **noun-phr**. Thus, an article is likely to be the most highly activated word and therefore selected and emitted. After an article is emitted the cursor is advanced to **np-2**, and so on. Advancing cursors is described in Section 5.5.

In this way constructions 'shunt' activation to words which should appear early in the input. FIG has no central process which plans or manipulates word order. Each construction simply activates nodes which it 'thinks' currently are relevant. In this sense word order is emergent.

5.4 Optional Constituents

When building a noun-phrase a generator should emit an adjective if semantically appropriate, otherwise it should ignore that option and emit a noun next. FIG does this without additional mechanism.

To see this, suppose "the" has been emitted and the cursor of **noun-phr** is on its second constituent, **np-2**. As a result adjectives get activation, via **np-2**, and so to a lesser extent do nouns via **np-3**. There are two cases: If the input includes a concept linked (indirectly perhaps) to some adjective, that adjective will receive activation from it. In this case the adjective will receive more syntactic activation than any noun does, and hence have more total activation, so it will be selected next. If the input does not include any concept linked to an adjective, then a noun will have more activation than any adjective (since only the noun receives semantic activation also), and so a noun will be selected next.

Most generators use some syntax-driven procedure to inspect semantics and decide explicitly whether or not to realize an optional constituent. In FIG, the decision to include or to omit an optional constituent (or adjunct) is emergent — if an adjective becomes highly activated it will be chosen, in the usual fashion, otherwise some other word, most likely a noun, will be.

5.5 Updating Constructions

Recall that FIG, after selecting and emitting a word, updates activation levels to represent the new state. In particular, it must advance the cursors of constructions as their constituents are completed [4]. Why is a separate update mechanism necessary? Most generators simply choose a construction and then 'execute' it straightforwardly. However, in FIG no construction is ever 'in control.' For example, one construction may be strongly activating a verb, but activation from other constructions may 'interfere,' causing an adverbial, for example, to be emitted instead. Therefore, in FIG constructions need feedback on what words have been output.

The difference between obl and opt constituents is whether or not the update mechanism can skip over them. (Since, for example, if there are no adjectives, the cursor of **noun-phr** should not remain stuck forever at the second constituent.) More than one construction may get updated after a word is output. For example, emitting a noun may cause updates to both the **prep-phr** construction and the **noun-phr** construction. Constructions which are 'guiding' the output should be scored as more relevant, so the update process adds activation to those constructions whose cursors have changed. It also sets a floor under their activation levels. After the last constituent of a construction has been completed, the cursor is reset and the floor is removed.

This type of bottom-up influence on constructions models an important factor affecting the syntactic form of utterances. [Bock and Warren 85] has shown that people can realize, after emitting some words, that in order to continue they must use a certain construction, for example, a passive. Similarly, FIG may output some words without any syntactic plan, then, based on the words output, the update mechanism will activate appropriate constructions, and those constructions will henceforth help guide production.

[3] For unordered construction the weight on all construction-constituent links is uniform and unchanging.

[4] Determining when a constituent is complete is not trivial. The current implementation uses a simple matching process, using the 'triggers' or each constituent (the third atoms in their descriptions).

5.6 No Instantiation or Binding

Most generators employ special mechanisms for instantiation and binding. One thing these are used for is handling multiple copies, for example, several noun phrases, or several instances of "a" in a single sentence. A connectionist system must also address this issue. An example of a problem that might occur otherwise is: in the case where several words of a category are highly activated, a node linked to all of them would receive more activation than when only one such word were active. For example **noun-phr** might receive activation from many words of category **noun**.

For this reason FIG uses a special rule for activation received across inherited links: the maximum (not the sum) of these amounts is used. For example, this rule applies to the 'maximal' links from nouns to **noun-phr**, which means that **noun-phr** effectively 'ignores' all but the most highly activated noun.

An earlier version of FIG handled this differently: by making copies of words and constructions. For example, it would make a copy of **noun-phrase** for each noun-expressible concept, and bind each copy to the appropriate concept, and to copies of **a-w** and **the-w**. Once I had started using instances and binding, it seemed natural to use those mechanisms for other problems (notably slots and cases). This approach worked, but it meshed so poorly with the basic activation-flow mechanism that I went back to a more pure spreading activation model. I conjecture that everything which seems to require 'instantiation' or 'binding' can be handled better by an appropriate refinement to the spreading activation mechanism.

5.7 Extended Example

This section describes how FIG produces "the old woman went to a stream to wash clothes." For this example the input is the set of nodes **ikuc1**, **old-womanc1**, **sentakuc1**, **kawac1**, and **pastc**. These nodes are linked to each other as follows: **ikuc1**'s **agentr** is **old-womanc1**, its **purposer** is **sentakuc1**, and its **destr** is **kawac1**; and **old-womanc1**'s **pragroler** is **topicc**. The concepts here have Japanese names because the input is the result of parsing the Japanese sentence "obaasan wa kawa e sentaku ni ikimashita," (old-woman TOPIC stream DEST wash-clothes PURPOSE go-POLITE-PAST).

Initially, each node of the input has 12 units of activation. Figure 6 shows the activation levels of selected nodes after activation flows, before any word is output. At this point the most highly activated word node is **the-w**, thus it will be selected and emitted first. The major source of activation for **the-w** is the first constituent of **noun-phr**, **np-1** (shown in capitals to indicate that it is the constituent currently under the cursor.) **np-1** receives energy from **noun-phr**, **noun-phr** receives most of its activation from **old-womanw**, which receives activation from **old-womanc1** and from **noun**. **noun** is activated, among other reasons, by the first constituent of **subj-pred**.

One construction not mentioned previously is **back-forep**, the construction responsible for putting adverbials of time and place at the beginning of a sentence. This construction has no effect on this sentence, since there are no concepts present expressible in this way.

After "the" is emitted **noun-phr** becomes even more highly activated and its cursor is moved to **np-2**. The most highly activated word becomes **old-womanw**, largely due to activation from **np-3**.

After "old woman" is emitted **noun-phr** is reset – that is, the cursor is set back to **np-1** and it thereby becomes ready to guide production of another noun phrase. Also, now the cursor on **subj-pred** advances to **sp-2**. As a result verbs, in particular **go-w**, become highly activated.

go-w is selected. Because **pastc** has more activation than **presentc** etc., **go-w** is inflected and emitted as "went." After this **subj-pred**'s cursor advances to its third constituent. At the same time, **go-p**'s cursor advances to its second constituent, thus it activates directional particles, although it happens that there is no semantic input to any such word. The most highly activated words are prepositions, due to activation from the first constituent of **prep-phr**, which in turn receives its energy from **sp-3** and to a lesser extent from **gp-3**. Of the various prepositions, **to1w** receives the most activation from **directionr**, which also receives its activation from the third constituent of **go-p**.

After "to" is emitted, the cursor of **prep-phr** is advanced. The key path of activation flow is now from the second constituent of **prep-phr** to **noun** to **streamw** to **noun-phr** to **article** to **a-w**. Thus "a" is emitted.

Then the cursor of **noun-phr** advances and "stream" is emitted. It is "stream" rather than some other noun because **streamw** is linked to **kawac**, **kawac** fills a **destinationr** relation, and **destinationr** is listed in **gp-3**.

At this point the cursor of **go-p** is on **gp-4**. From this constituent activation flows to **purpose-clause**, and in due course "to" and "wash clothes" are emitted.

In this way FIG has produced: "the old woman went to a stream to wash clothes." All the nodes of the input having been expressed, FIG ends.

6 Implementation

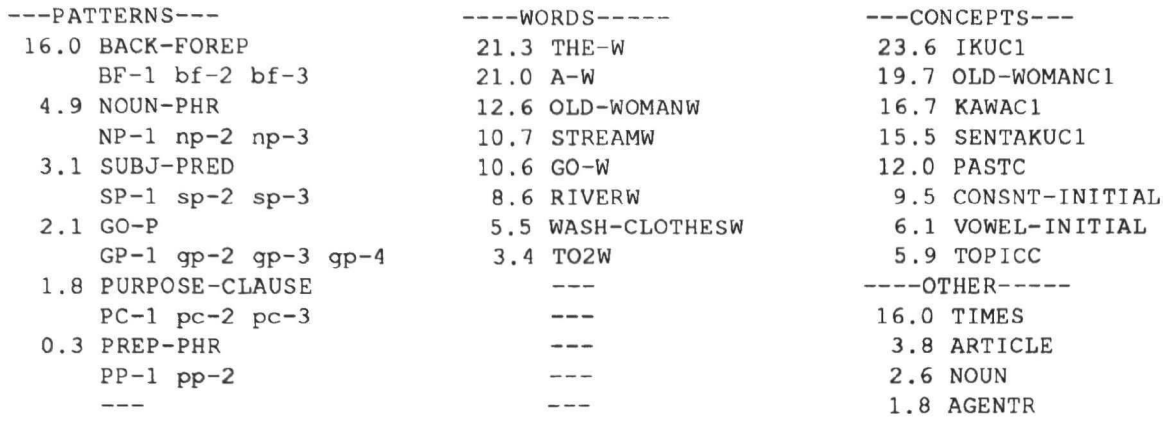


Figure 6: Activation Levels of Selected Nodes After Activation Flow

FIG currently has six types of nodes: concepts, relations, words, constructions, constituents, and instances of concepts. They are distinguished for clarity and efficiency but not for activation flow. The cursor update process and semantic update process (not described further) do, however, examine node types; and of course only words are selected and emitted.

Although links have been differentiated above in terms of their intended meaning, activation flows across all links in the same way, except: 1. there are **english** and **japanese** links; no activation flows across the ones for the language not in use, 2. the weights of links from constructions to constituents are modified by the mask according to the cursor, and 3. inhibit links transmit negative activation.

In accordance with the intuition that a word is not truly appropriate unless it is both syntactically and semantically appropriate, the activation level for words is given by the product (not the sum) of incoming syntactic and semantic activation. 'Syntactic activation' is activation received from constituents and syntactic categories.

Currently FIG has 284 nodes and about 600 links. Before each word choice, activation flows until the network settles down, with cutoff after 8 cycles. This takes about .15 seconds on average (simulating parallel activation flow on a Symbolics 3670), thus FIG outputs words faster than a human speaker.

The correct operation of FIG depends on having correct link weights. This is not a major problem. Many of the link weights are uniform. For example, all links from syntactic categories to their members have weight .8, all 'inhibit' links have weight .7, and so on. Many of the others have a rationale: for example, the link from **np-1** to articles has relatively high weight because articles get very little activation from other sources. No single weight is meaningful; the way it functions in context is. For example, the exact weight of the link from the first constituent of **subj-pred** to **noun** is not crucial, as long as the product of it and the weight on the **agentr** relation is appropriate.

Also crucial in generation is the flow of activation through the network structures encoding world-knowledge. World knowledge is also used when monitoring the output and updating activation levels. For details see [Ward 88a].

FIG is, of course, extensible. Adding new concepts, words or constructions is generally straightforward; they can be encoded by analogy to similar nodes, and usually the same link weights suffice. Occasionally new nodes and links interact in unforeseen ways with other knowledge in the system, causing other nodes to get too much or too little activation. In these cases it is necessary to debug the network. Sometimes trial-and-error experiments are required, but often the acceptable range of weights can be determined by examination. This is a kind of back-propagation by hand; it could doubtless be automated.

Besides just increasing the amount of knowledge in FIG's network, I would like to make it model human speech errors and to use its grammatical knowledge structures for parsing also.

7 Summary

FIG's treatment of syntax is connectionist and novel. It relies heavily on parallelism and emergents. It does not build up any syntactic structures, nor even make explicit syntactic choices. The only explicit choices needed are the successive choices of words. It is also novel in that the network representations of linguistic knowledge affect word choice and order directly, rather than just affecting a parse tree. Thus the implementation corresponds clearly and directly to the knowledge-level theory.

This treatment of syntax works well with the rest of FIG — all types of knowledge are well integrated and interact freely at run time. I have explained elsewhere how this is important for: accurate and flexible word choice [Ward 88a], producing natural-sounding output for machine translation [Ward 89a], and modeling the key aspects of the human language production process [Ward 89c].

This work is not traditional linguistics, artificial intelligence, or connectionism, but uses techniques from all three fields. I have presented a syntactic mechanism which is compatible with intuitions about syntactic knowledge and also with connectionist processing. I hope this will stimulate further work in empirical computational linguistics, modeling human language production, and building and useful parallel generation systems.

References

[Baars 80] Baars, Bernard K,. The Competing Plans Hypothesis: an heuristic viewpoint on the causes of errors in speech, in *Temporal Variables in Speech,* Hans W. Dechert and Manfred Raupach (eds.), Mouton, 1980.

[Bock and Warren 85] Bock, J. K., and Richard Warren, Conceptual Accessibility and Syntactic Structure in Sentence Formulation, *Cognition* 21, 1985.

[Dell 86] Dell, Gary S., A Spreading Activation Theory of Retrieval in Sentence Production, *Psychological Review* 93, 3, 1986.

[Fillmore 88] Fillmore, Charles, *On Grammatical Constructions*, course notes, UC Berkeley Linguistics Department, 1988.

[Fillmore et al 89] Fillmore, Charles, Paul Kay, and M. C. O'Connor, Regularity and Idiomaticity in Grammatical Constructions: The Case of Let Alone, *Language* 64, 3, pp 501-538, 1988.

[Gasser 88] Gasser, Michael, A Connectionist Model of Sentence Generation in a First and Second Language, PhD Thesis, also Technical Report UCLA-AI-88-13, Los Angeles, 1988.

[Kalita and Shastri 87] Kalita, Jugal, and Lokendra Shastri, Generation of Simple Sentences in English Using the Connectionist Model of Computation, *9th Cognitive Science Conference,* Erlbaum, 1987

[McDonald 1983] McDonald, David, Description Directed Control: Its Implications for Natural Language Generation, *Computers and Mathematics with Applications*, 9, 1 1983.

[Stemberger 85] Stemberger, J. P., An Interactive Activation Model of Language Production, in *Progress in the Psychology of Language, Volume 1*, Andrew W. Ellis, ed., Erlbaum, 1985.

[Stolcke 89] Stolcke, Andreas, Processing Unification-based Grammars in a Connectionist Framework, *11th Cognitive Science Conference,* Erlbaum, 1989.

[Ward 88a] Ward, Nigel, Issues in Word Choice, *COLING-88*, Budapest, August 1988.

[Ward 88b] Ward, Nigel, An Open Design for Generation, *Proceedings of the AAAI Workshop on Text Planning and Realization*, St. Paul, MN, August 1988.

[Ward 89a] Ward, Nigel, Towards Natural Machine Translation, Proceedings of the EIC Workshop on Artificial Intelligence, Published as Technical Research Report AI89-29~37, Institute of Electronics, Information, and Communication Engineers, Tokyo, June 1989.

[Ward 89b] Ward, Nigel, On the Ordering of Decisions in Machine Translation, *Proceedings of the National Conference of the Japanese Society for Artificial Intelligence*, Tokyo, July 1989.

[Ward 89c] Ward, Nigel, Capturing Intuitions about Human Language Production, *Proceedings, Cognitive Science Conference,* Ann Arbor, August 1989.

Harmonic Grammar - A formal multi-level connectionist theory of linguistic well-formedness: Theoretical foundations

Géraldine Legendre[1,2], Yoshiro Miyata[1,3,4], Paul Smolensky[1,3,4]

[1]*Institute of Cognitive Science,* [2]*Department of Linguistics,* [3]*Department of Computer Science,*
& [4]*Optoelectronic Computing Systems Center, University of Colorado at Boulder*

Abstract

In this paper, we derive the formalism of *harmonic grammar*, a connectionist-based theory of linguistic well-formedness. Harmonic grammar is a two-level theory, involving a low level connectionist network using a particular kind of distributed representation, and a second, higher level network that uses local representations and which approximately and incompletely describes the aggregate computational behavior of the lower level network. The central hypothesis is that the connectionist well-formedness measure *harmony*[1] can be used to model linguistic well-formedness; what is crucial about the relation between the lower and higher level networks is that there is a harmony-preserving mapping between them: they are *isoharmonic* (at least approximately). In a companion paper (Legendre, Miyata, & Smolensky, 1990; henceforth, "LMS_1"), we apply harmonic grammar to a syntactic problem, unaccusativity, and show that the resulting network is capable of a degree of coverage of difficult data that is unparallelled by symbolic approaches of which we are aware: of the 760 sentence types represented in our data, the network correctly predicts the acceptability in all but two cases. In the present paper, we describe the theoretical basis for the two level approach, illustrating the general theory through the derivation from first principles of the unaccusativity network of LMS_1.

Introduction

Our starting point is the approach to connectionist cognitive modeling called the *subsymbolic paradigm* (Smolensky, 1988):

(1) **Hypotheses of the subsymbolic approach to cognitive modeling**

a. There are two important levels for cognitive modeling.

b. At the lower level, the natural description of the cognitive architecture is as a massively interconnected network of simple parallel numerical processing units: call this LNet (Lower level Network).

c. In LNet, elements of the problem domain (e.g., in syntax, words and phrases) are represented not by individual units, but as distributed patterns of activity; a given unit in LNet has no semantic interpretation by itself: it plays a small part in the representation of many different elements.

d. When the representations and computational processing of LNet are described at the higher level of the semantically meaningful activity patterns, we get descriptions of the cognitive architecture at the higher level. Such descriptions will often be approximate, idealized, or incomplete.

e. Unlike the lower level, descriptions of the higher level are not computationally uniform. Some of these descriptions involve symbolic computation with hard rules. Others involve local connectionist networks, in which individual units have semantic interpretations corresponding to those of the patterns of LNet. Such networks will be called HNets (Higher level Networks).

f. The symbols and rules of symbolic accounts correspond in LNet to patterns of activity and to the aggregate effects of groups of connections on these patterns of activity.

g. An important goal of connectionist modeling is to develop LNets supporting higher level descriptions that are simultaneously (i) sufficiently close to symbolic cognitive theory to explain the successes of symbolic accounts, yet (ii) sufficiently different to improve upon these successes.

The central idea of harmonic grammar is to start by partially specifying a LNet for a domain of linguistic interest, and then, rather than fully specifying and simulating it, as is conventionally done in connectionist modeling, to embody the most important aspects of LNet in a higher-level net HNet. This model, or rather a notational variant of it, HNet′, is what gets simulated. HNet′ (or, equivalently, HNet) is interpreted as grammar fragment expressing linguistic regularities via *soft rules*. Whereas symbolic rules of well-formedness have the form (2a), the soft rules of harmonic grammar have the form (2b).

(2) a. Condition X must never be violated in well-formed structures.

b. If Condition X is violated, then the well-formedness (harmony) of the structure is diminished by C_X.

The status of the two networks LNet and HNet are rather different. The level of LNet is presumably closer to the neural level, and therefore provides a more appropriate model for questions related to neurolinguistics (although the problematic relationship between connectionist and neural models, emphasized in Smolensky, 1988, suggests caution here). For language acquisition and real-time language processing models, as well, LNet would presumably be the more appropriate network. But for grammar, it is HNet that is the focus of attention.

This paper procedes as follows. We begin with a number of technical preliminaries which, after brief introduction of the linguistic problem of unaccusativity, motivate LNet, a partially specified lower level model for the unaccusativity data. We then derive a corresponding higher level model, HNet, and then its notational variant, HNet', which is the model discussed in LMS_1. HNet' allows standard connectionist learning to automatically extract from the data the constants C_X in the soft rules (b) of harmonic grammar. We close by summarizing the methodology and identifying some of its novel features.

Technical preliminaries

The subsymbolic approach outlined in (1) and its application to the domain of language presents the following research challenges, among others:[2]

(3) A. Representation:
   1. Develop a formalism for higher level description of distributed representations: a calculus of patterns of activity (it is these patterns that correspond to symbols; (1f))
   2. Apply this calculus to the representation of constituent structure

B. Processing:
   1. Develop a formalism for higher level description of connectionist processing: a calculus of the aggregate effects of groups of connections on patterns of activity (it is these effects that correspond to rules; (1f))
   2. Apply this calculus to the processing of constituent structure

The next four subsections successively address these four problems: A.1, A.2, B.1, and B.2.

A calculus of patterns of activity

A natural "calculus of patterns of activity" is straightforward: vector calculus, where the vectors are the lists of activation values for the units. The central idea of distributed representation (1c) can be stated very simply: it is activity vectors (not, e.g., individual units) that have semantic interpretations, i.e., interpretations as elements of the problem domain (the kind of information that is represented by symbols in the symbolic paradigm).

If different symbols are represented by different patterns of activity over the same set of units, as hypothesized in (1c), how is it possible to represent several such symbols at once? Vector calculus suggests a simple answer: by *superimposing*, i.e, adding together, the vectors representing the individual symbols. In the symbolic paradigm, structures are formed by some kind of (e.g., string or tree) concatenation of their constituents; in the subsymbolic approach, patterns combine by superposition rather than concatenation. This principle is discussed at some length in Smolensky (1986b), where an important consequence is derived: to a given network using distributed superpositional representations, there corresponds another network using local representations which provides a higher level description of the distributed network. The formal relation between the lower and higher level models can be thought of as a "rotation of the coordinates" in the activation space, so that the new coordinate axes lie along the directions of the distributed patterns with semantic interpretation; alternatively, we can think of this relation as changing variables from the lower level variables — units' activation values — to higher level variables — the strength of semantically interpretable patterns.[3]

Vectorial representation of constituent structure

How can simple vector addition replace concatenation? One basic problem that immediately suggests itself is that the former is a commutative operation, while the latter is not; e.g., `concat(a,b) = ab ≠ ba = concat(b,a)`, while **sum(a,b) = a+b = b+a = sum(b,a)**. A solution to this and related problems, called tensor product representations, is formalized and analyzed in Smolensky (in press 1). The first step is to recognize that vector superposition really represents *conjunction* rather than *concatenation*, and that concatenation, and other structure-building operations, can be achieved through conjunction together with *filler/role decompositions*. In such decompositions, a structure, e.g. `abc`, is described as the conjunction of an unordered set of propositions of the form, *structural role r is filled by f*, which are denoted by the *filler/role bindings f/r*; thus, e.g., `abc` is identified with the conjunction of the filler-role bindings $\{a/r_1,\ c/r_3,\ b/r_2\}$. The vector representing `abc`, under this filler/role decomposition, is $\mathbf{abc} = \mathbf{a/r_1+c/r_3+b/r_2}$ The vectors representing filler/role bindings, e.g. b/r_2, are constructed from vectors representing the unbound fillers and vectors representing unbound roles, e.g., b and r_2, by an operation from vector calculus called *the tensor product*: $\mathbf{b/r_2} = \mathbf{b} \otimes \mathbf{r_2}$. The tensor product is similar to the outer product of matrix algebra, e.g. $(x, y, z) \otimes (\alpha, \beta) = (x\alpha, x\beta, y\alpha, y\beta, z\alpha, z\beta)$ except that the result is interpreted not as a matrix but as another vector; the resulting vector can in turn be used recursively in further products, allowing recursive representations employing higher-order tensors. Smolensky (in press 1) analyzes these ideas — decompositions of structures into conjunctions of filler/role bindings, the superpositional representation of conjunction, and representation of filler/role variable bindings via the tensor product — and shows that together they formalize and generalize a number of previous connectionist approaches to representing structure, and that they represent structured data in a way that naturally permits the usual features of connectionist processing, e.g., massively parallel (and structure-sensitive) associative processing, graceful degradation, and statistical learning.[4]

Thus, for the purposes of this paper, we assume:

(4) **Tensor product representation of structure**
At the lower level, in LNet, a structure s is represented by the activation vector $\mathbf{s} = \sum_\alpha \mathbf{c}_\alpha$ where each $\mathbf{c}_\alpha$ represents a constituent of s, which is a filler/role binding f_α/r_α with respect to some filler/role decomposition of s; the constituent vectors are $\mathbf{c}_\alpha = \mathbf{f}_\alpha \otimes \mathbf{r}_\alpha$, where $\mathbf{f}_\alpha$ and $\mathbf{r}_\alpha$ are activity vectors respectively representing f_α and r_α (possibly recursively defined as tensor product representations themselves).

A calculus for connectionist processing

Viewed at the lowest level, connectionist processing is the spread of numerical activation by some set of numerical equations in which the connection strengths enter as parameters. A calculus of the aggregate effects of connection strengths on patterns of activity relies on the idea that these activation equations are trying to achieve some characterizable end product: a pattern of activity that encodes a set of inferences which is justifiable from some notion of statistical inference. Smolensky (1983, 1986a) developed such a higher level description, deriving from a well-defined statistical inference problem a measure called the *harmony function H* whose global maximum constitutes the solution to the inference problem. In a large variety of connectionist models, the activation functions turn out to be implementing local maximization of this function, which can be written very simply:

(5) $H(\mathbf{a}) = \sum_{i,j} a_i w_{ij} a_j = \mathbf{a}^\mathbf{T}\mathbf{W}\mathbf{a}$

where $\mathbf{a} = (a_1, a_2, \ldots)$ is the total activity vector of the network, and $\mathbf{W} = \{w_{ij}\}$ is the matrix of connection weights. (The negative of H is often referred to as "energy"; Hopfield, 1982; Hinton & Sejnowski, 1983, 1986.)[5] In a variety of networks, including both feed-back and feed-forward architectures, each update of the units' activations will increase H (Cohen & Grossberg, 1983, Golden, 1986, 1988, Hopfield 1982, 1984, 1987, Smolensky, 1983, 1986a, Hinton & Sejnowski, 1983, 1986.) Thus for a major subset of connectionist models, it is appropriate to regard the goal of the spread of processing to be the creation of an activation vector that maximizes H. For the purposes of formulating approximate higher level accounts of connectionist processing, we will thus assume that such harmony maximization is what processing in LNet achieves, even though a more detailed lower level account might well need to consider conditions under which the network fails to actually find the global maximum of H. Thus for our purposes we assume that under suitable approximation or idealization, the following holds:

(6) **Principle of harmony maximization**
Given an input vector **i** and connection weights **W**, processing in LNet establishes activity vectors **h** and **o** over the hidden and output units, respectively, that maximize $H(\mathbf{a})$, where $\mathbf{a} = (\mathbf{i}, \mathbf{h}, \mathbf{o})$ and the harmony function H is defined in (5).

Given the connection between H and statistical inference, this principle can be interpreted as follows: the network draws a set of inferences that provides a best fit to the input data and the statistical constraints embodied in **W** (e.g., the "Best Fit Principle" of Smolensky, 1988;Golden 1988). The value of H that is achieved in this maximization process is a quantitative measure of the degree to which it is possible to meet the statistical constraints in **W** by appropriately processing the given input data. This motivates the following central assumption of harmonic grammar:

(7) The H value achieved in processing an input is a quantitative measure of that input's well-formedness. An informant's judgements of acceptability of sentences can be modeled as a monotonic function f of the harmony values achieved in processing those sentences (the higher the harmony, the more acceptable).

In this paper, we take f to be the particular monotonically increasing function $f(x) = (1+e^{-x})^{-1}$: a logistic; the same methodology could be carried out for any other choice of (differentiable) f.

Connectionist processing of constituent structure

Now we bring together (4) and (5), assuming that the activity vector **a** in (5) is the structure representation **s** in (4). Then (6) states that processing in LNet maximizes:

(8) $H(\mathbf{s}) = \mathbf{s}^T \mathbf{W} \mathbf{s} = \sum_\alpha \mathbf{c}_\alpha^T \mathbf{W} \sum_\beta \mathbf{c}_\beta = \sum_\alpha H_\alpha + \sum_{\alpha,\beta} H_{\alpha,\beta}$

Here, the sum $\sum_{\alpha,\beta}$ is understood to include *one* term for each pair of distinct constituents $c_\alpha \neq c_\beta$ in s, and the first- and second-order harmony terms are defined by:

(9) a. $H_\alpha = \mathbf{c}_\alpha^T \mathbf{W} \mathbf{c}_\alpha = \sum_{i,j} (\mathbf{c}_\alpha)_i w_{ij} (\mathbf{c}_\alpha)_j$

b. $H_{\alpha,\beta} = \mathbf{c}_\alpha^T \mathbf{W} \mathbf{c}_\beta + \mathbf{c}_\beta^T \mathbf{W} \mathbf{c}_\alpha = \mathbf{c}_\alpha^T [\mathbf{W}+\mathbf{W}^T] \mathbf{c}_\beta = \sum_{i,j} (\mathbf{c}_\alpha)_i [w_{ij}+w_{ji}] (\mathbf{c}_\alpha)_j$

For example, if s = abc is decomposed as $\{a/r_1, c/r_3, b/r_2\}$, then $H(abc) = H_{a/r_1} + H_{b/r_2} + H_{c/r_3} + H_{a/r_1,\ b/r_2} + H_{a/r_1,\ c/r_3} + H_{b/r_2,\ c/r_3}$. H_α is the internal harmony associated with constituent c_α, and $H_{\alpha,\beta}$ is the pairwise harmony arising from combining the constituents c_α and c_β in the same structure. Note that terms depending on more than two constituents cannot arise because H is quadratic.

Now if our specification of LNet were sufficiently complete that we knew the weight matrix W and the activity patterns representing the constituents $\mathbf{c}_\alpha$, we could compute each of the terms in (8) through the equations (9). This would amount to computing at the lower level. Alternatively, *we can operate with the harmony terms in (8) directly*, treating each H_α and $H_{\alpha,\beta}$ as an independent variable; this is computing at the higher level. Exploitation of this alternative is a central innovation of harmonic grammar.

The distinction between computing at the lower and higher levels can be viewed as follows. Equation (5) expresses the harmony as a function of the *lower level variables*: activities of individual units and strengths of individual weights. Equation (8), on the other hand, expresses the harmony as a function of the *higher level variables*: the constituents in the structure. The derivation of (8) from (5) [and (4)] amounts to a *change of variables* from the lower level variables associated with units and connections to the higher level variables associated with constituents. These higher level variables will shortly be used to define HNet.

We conclude these preliminaries with a few remarks.

(10) a. In a given problem, some of the constituents c_α will be "inputs," others will be "outputs," and still others will be "hidden" constituents. E.g., a sentence interpretation model might take as input constituents a string of words, might produce as output constituents the elements of some meaning representation, and might fill in as hidden constitents, say, non-terminal nodes in a parse tree. The "inferred" constituents — the hidden and output constituents — are determined through the principle of harmony maximization (6): they are the choice of c_αs that maximize H in (8).

b. The higher level harmony values H_α and $H_{\alpha,\beta}$ will later be interpreted as the constants in the soft rules of (b). They will be computed by a numerical fit to data. When is this parameter fitting exercise appropriately constrained? Note that, since H in (8) is quadratic in the constituents, the number of different terms that may appear on the right-hand-side of (8) scales as $(\#fillers)^2$, while the number of possible structures on the left-hand-side of (8) scales roughly as $(\#fillers)^{\#roles}$. Thus, as the number of roles in the structure increases, the formalism rapidly becomes more and more constrained, and the significance of good parameter fit becomes increasingly more meaningful.

c. Is it correct to treat the higher level harmony variables H_α and $H_{\alpha,\beta}$ as independent? This clearly depends on the structure assumed in LNet. Crudely speaking, if there are many more lower level variables than higher level variables, it is likely that any set of values for the higher level variables H_α and $H_{\alpha,\beta}$ can be achieved by some choice of the lower level variables, the representations $\mathbf{c}_\alpha$ and weights W. On the other hand, if it is possible to identify some strong constraints on the lower level variables — e.g., if a large number of different constituents were constrained to be represented as different activity patterns over a much smaller number of units — then, effectively, there might be fewer lower level variables than higher level ones, so that the space of possible values for the higher level variables might be genuinely constrained by the fact that they are derived from the lower level.

A sample application: Unaccusativity in French

Unaccusativity

Since the problem of unaccusativity is discussed in some detail in LMS_1, we are extremely brief here. In many languages, in particular French, intransitive verbs divide into two classes: *unergatives* and *unaccusatives*, which yield different acceptability judgements in certain syntactic environments called *diagnostic contexts* (here, simply "contexts"). Consider, for example, the sentence *La glace est facile à faire fondre* "Ice is easy to make melt." Here, the diagnostic context is Object Raising or "OR," which is a sentence frame _____ *est facile à faire* _____ having two slots; the first, "argument," slot is filled by the NP *La glace*, and the second, "predicate," slot is filled by the intransitive verb *fondre*. The data of LMS_1 are 760 such French sentences, generated from four different diagnostic contexts, 143 different intransitive verbs, and arguments with varying semantic features. The pattern of acceptability judgements for these 760 sentences is quite complex. The acceptability patterns across different contexts of roughly half the verbs can be explained by a standard symbolic syntactic account which postulates that all the diagnostic contexts have well-formedness conditions requiring the argument to be a deep Direct Object of the predicate, and that each intransitive verb is marked in the lexicon as requiring its argument to be either a deep Subject (unergative verbs) or a deep Direct Object (unaccusative verbs). The other half of the data can only be explained by assuming that the acceptability reflects not only these "structural features" (deep Subject, Direct Object), but also semantic features of the argument and predicate. At the same time, these semantic features alone do not seem to be sufficient either; structural and semantic features are both required to explain these data.

LNet

Applying (1g) and (4) to the case at hand, we assume:

(11) a. Symbolic structural descriptions of sentences are approximate higher level descriptions of the patterns of activity in a lower level model LNet representing those sentences. In particular, the argument of an intransitive verb fills, among other roles, either the structural role of deep Subject or that of deep Direct

Object.

b. In particular, these patterns of activity can be approximated as tensor product representations based on a role filler/role decomposition exemplified as follows for the vector representing the Object Raising (OR) sentence *La glace est facile à faire fondre*:

La_glace_est_facile_à_faire_fondre =
La_glace/ARG + OR/CONTEXT + fondre/PRED + DIRECT_OBJECT/STRUCTURE =
A + C + P + S

That is, the vector representing a sentence can be approximated as the sum of four vectors, each of which represents a kind of constituent that is specially designed for the particular data under study: an argument A, a context C, a predicate P, and a "structure" (deep grammatical function) S (either Subject or Direct Object).

We will not further specify this partial description of LNet; in particular, we will not specify vectors representing the individual fillers and roles. In the most general case, these vectors may be presumed to be fully distributed, giving rise to a representation of sentences in which every unit is part of the representation of each constituent; it will not matter to our analysis whether this fully distributed case or a more localized special case obtains, e.g., one in which the four roles of (11b) are localized to disjoint regions of the network.

There is no particular point in drawing a picture of LNet; we need only imagine a large network holding a representation of the sentence as a pattern of activity which is the sum of four constituent patterns (each of which may well involve activation over the entire network), according to (11b). Three constituents of this vector — the argument, context, and predicate — are specified in the input: the surface word string of a given sentence. The fourth constituent — the structure feature — is not given in the input; it is "hidden" (10a), and must be inferred by the network through activation spread to maximize harmony. Following (7), the degree of acceptability of the sentence to the network is taken to be $f(H)$, where H is the harmony of this activation pattern, and f is the logistic function. In LNet, acceptability is a distributed property; there is no "output unit" giving the network's acceptability judgement.

Why do we assume the filler/role decomposition of (11b)? Because it is the simplest imaginable one with which to start. The remarkable success of the consequent model provides some evidence in favor of this very simple assumption. It should be made clear, however, that the methodology permits assuming a different filler/role decomposition of the sentences, and following it through to a corresponding higher level model HNet operating in terms of the different constituent filler/role pairs, just as we now do for the filler/role decomposition assumed in (11).

HNet

Combining (6), (7), (8), and (11b), we have:

(12) The acceptability of a sentence s consisting of the argument A, the context C, and the predicate P is:

$$\text{acceptability}(s) = f\,[\max_S H(\mathbf{A+C+P+S})]$$

where S ranges over the two possible structures, Subject and Direct Object, and

$$H(\mathbf{A+C+P+S}) = H_A + H_C + H_P + H_S + H_{AC} + H_{AP} + H_{AS} + H_{CP} + H_{CS} + H_{PS}$$

This equation for H involves a prohibitively large number of higher-level parameters H_α and $H_{\alpha\beta}$. We eliminate a great many of these parameters by appealing to a number of linguistically motivated constraints:

(13) a. H_A: assume all arguments in the sentences used are equally well-formed internally

b. H_P: assume all predicates in the sentences used are equally well-formed internally

c. H_S: assume the grammar has no intrinsic preference between deep Subjects and deep Direct Objects

d. H_{AC}: assume the grammatical restrictions on the diagnostic context can refer only to general semantic features of the argument (not, e.g., to specific NPs); we take these features to be VO (volitionality) and AN (animacy)

e. H_{AP}: assume the lexical entry for the predicate can only express preference for general semantic features of the argument (again, VO and AN)

f. H_{AS}: assume the grammatical preferences for semantic/structural correspondences of the argument can only depend on its general semantic features (VO, AN)

g. H_{CP}: assume the grammatical restrictions on the diagnostic context can refer only to general semantic features of the predicate (not, e.g., to specific verbs); we take these features to be TE (telicity) and PR (progressivity)[6]

h. H_{CS}: assume the grammatical restrictions on the diagnostic context can refer to the structure (Subject vs. Direct Object) of the argument

i. H_{PS}: assume the lexical entry for a predicate can include a structural preference, but not an absolute bias on grammaticality

Assuming these constraints to hold, and dropping H_A, H_P and H_S because they do not vary across sentences (13a-c), we can rewrite the harmony function:

(14) $$H(\mathbf{A+C+P+S}) = H_C + H_{VO,C} + H_{AN,C} + H_{VO,P} + H_{AN,P} + H_{VO,S} + H_{AN,S} + H_{C,TE} + H_{C,PR} + H_{C,S} + H_{P,S}$$

Now we recognize this as the harmony function of another network, HNet, illustrated in Figure 1. HNet uses a local

representation, with a single unit for each each context, argument feature, predicate feature, structural feature, and individual predicate. The units in HNet correspond to patterns in LNet; HNet is a higher level network that is *isoharmonic* to LNet: the harmony of corresponding states in the two models are the same.

This network can be used to compute acceptability as follows. A given sentence is represented over the input units. We activate which ever of the hidden units gives the greatest harmony; this can be achieved by having the two units compete so that the unit with the greater net input wins: the net input to each hidden unit is precisely the contribution to the total harmony that that hidden unit would make if it were to have activity value 1. The hidden units thus are a little "winner-take-all" group in which the winning unit gets activity value 1, and the other, 0. Now we compute the harmony H of the network as a whole using (5) (with the variables H_α and $H_{\alpha\beta}$ now playing the roles in (5) of the weights W). Putting this value H into f, we get the acceptability $f(H)$, following (7).

The weights in this network are the harmony values H_α and $H_{\alpha\beta}$ of (10); from the point of view of the original lower level model, each of these weights represents the aggregate harmony of a set of weights and activity vectors, as indicated in (9). We'd like to work backwards from the data to infer what these aggregate values must be in order to produce the observed well-formedness pattern, but training the harmony values that are distributed throughout this network is not straightforward. A few simple modifications in the network, though, will fix this.

HNet′

In order to perform standard supervised learning from the data, we now create a network HNet′ that is precisely equivalent to HNet, but which possesses a single output O unit which explicitly computes acceptability. The main trick, illustrated in Figure 2, is simple: replace the connection between input units α and β of HNet, carrying weight $H_{\alpha\beta}$, by a conjunction unit $c_{\alpha\beta}$ whose activity is the product $a_\alpha a_\beta$, and connect $c_{\alpha\beta}$ to O with a connection of strength $H_{\alpha\beta}$. Then the contribution to O's net input coming from $c_{\alpha\beta}$ is $a_\alpha a_\beta H_{\alpha\beta}$, which is just the amount of harmony in HNet contributed by the original connection between α and β. Thus the total net input to O is the total harmony H. If O uses f to transform its net input to its activation value, then its value is exactly the acceptability judgement of Figure 1.

The only remaining step concerns the hidden units. In defining HNet′, the trick of replacing HNet connections with conjunction units should not be applied to the connections to the hidden units; these connections just stay as in HNet. As noted above, the net input to each hidden unit is precisely the contribution to the total harmony that that hidden unit would make if it were to win the competition. Suppose we define the activation functions of HNet′'s hidden units so that their activity values prior to competition are just equal to their net inputs; this is also their contribution to H. To maximize H, we let these two units compete so that the one with the higher activity value retains its value, while the other has its value set to zero. Now, the hidden units send their activation values to O (along connections with strength 1); the hidden unit capable of contributing the greatest harmony sends that harmony value up O, whose net input now includes the correct contribution from the hidden units, namely, the harmony arising from picking the structural feature that maximizes harmony.

The network HNet′ we have just defined, shown in Figure 3, is precisely the network used in LMS_1; and we have completed its derivation from basic principles. This network can now be trained using the appropriate form of standard back propagation (Rumelhart, Hinton, & Williams, 1986), as described in LMS_1. Note that the assumption of independence of the higher level variables, discussed in (10c), is relevant here, if the training procedure incorporates no constraints between weights. Note also that the "learning" in HNet′ is not a plausible model of language acquisition (for one thing, positive and negative data are crucial); "learning" in HNet′ is purely a computational procedure for parameter fitting, a formal trick for automating (a particularly nasty) part of the job of the harmonic grammarian: determining the numerical constants C_χ in the soft rules (b). Presumably, a plausible model of real language acquisition would operate at the lower level, in LNet, rather than in HNet′.

Summary of the methodology

The methodoloy of harmonic grammar exemplified above can be summarized as the following series of steps.

(15) a. Choose a filler/role decomposition for the structures whose well-formedness is to be accounted for.
b. Postulate a lower level model LNet using a tensor product representation with this filler/role decomposition.
c. Take the formula (5) for the harmony of LNet in terms of its weights and activities, and change variables ...
d. ... to get a formula (8) for the harmony as function of the constituents of the structure being represented; this function involves aggregated harmony values indexed by pairs of constituents; treat these values as independent high level variables.
e. Prune the number of these variables by appealing to linguistic constraints (at least) until the number of variables is considerably fewer than the number of data points to be accounted for.
f. Embody the resulting harmony function as a local connectionist network HNet whose connection strengths are the high level harmony variables.
g. Create HNet′ by adding to HNet an output unit that explicitly computes the harmony and corresponding acceptability value by means of additional conjunctive units and winner-take-all linear hidden units.
h. Train HNet′ using more-or-less standard connectionist supervised learning.

i. Interpret HNet' as embodying soft grammatical and lexical rules.

j. Analyze these rules for new linguistic insight into the original linguistic problem.

Step (15j) is the subject of current research.

The method exhibits the following novel features:

(16) a. It is founded on a distributed lower level connectionist model that is only partially specified.

b. It operates primarily through a higher level formalism that approximately describes certain aspects of the aggregate behavior of the lower level network in terms of another, local, connectionist network.

c. The grammatical and lexical rules of the formalism are soft, and represent a set of quantified tendencies; but the model is fully formal, in that it makes precise predictions (even graded ones) of acceptability or well-formedness.

d. The strength of the soft rules is determined automatically from the data.

e. Existing linguistic knowledge plays the important role of constraining the form of the grammar.

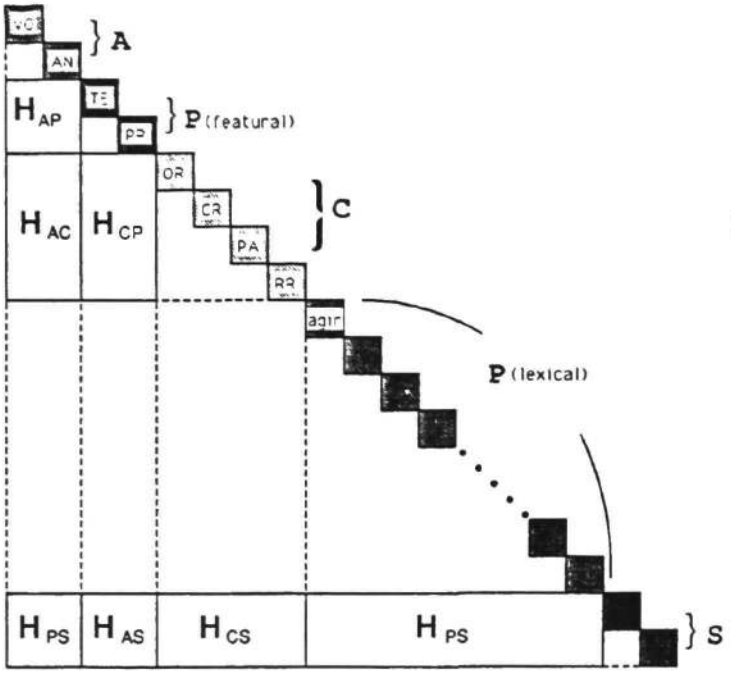


Figure 1.

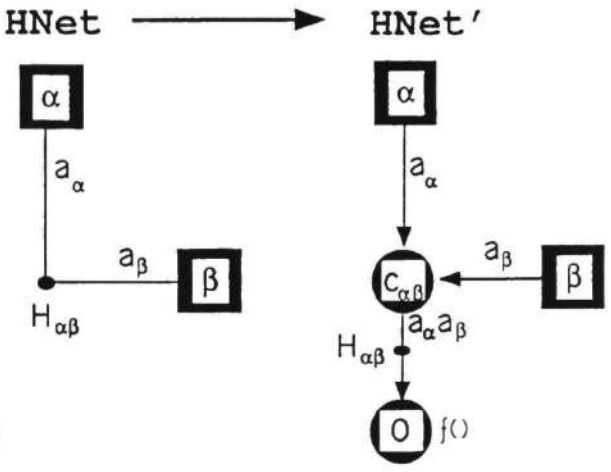


Figure 2.

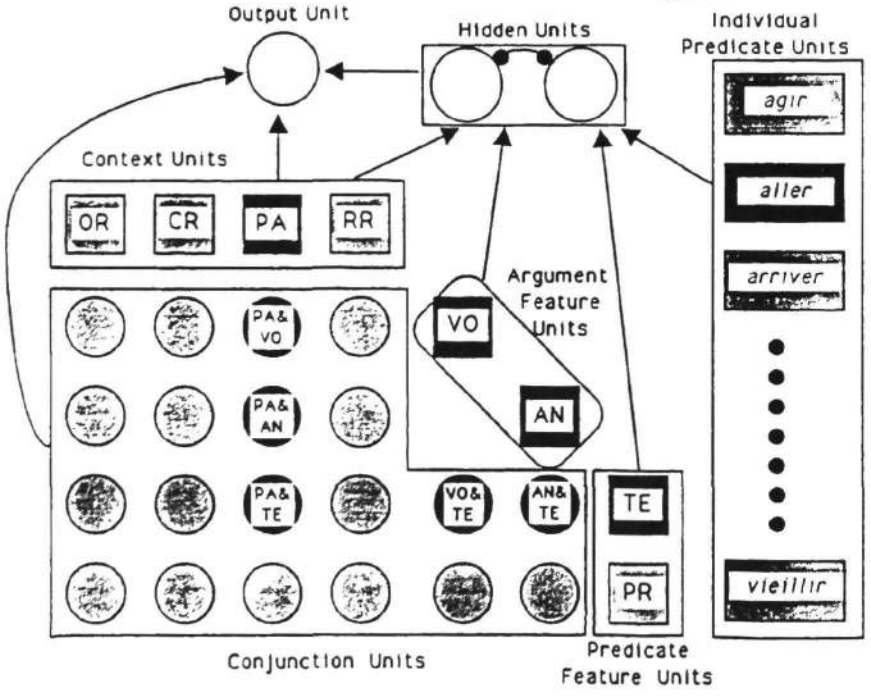


Figure 3.

Acknowledgements

Warm thanks to Alan Prince, for very helpful discussions, and especially, a great pot of chicken soup and the term "isoharmonic." This work owes its existence to Mike Mozer, who failed to convince us not to do it. Thanks also to Jim Martin for his valuable comments on an earlier version. This work has been supported by NSF grants IRI-8609599 and ECE-8617947 to PS, by a grant to PS from the Sloan Foundation's computational neuroscience program. PS (in part) and YM have also been supported by the Optical Connectionist Machine Program of the Center for Optoelectronic Computing Systems, which is sponsored in part by NSF/ERC grant CDR-8622236 and by the Colorado Advanced Technology Institute, an agency of the State of Colorado. GL has been supported in part by a Junior Faculty Development Award from the Council on Research and Creative Work, University of Colorado, Boulder. The authors are listed in alphabetical order.

Footnotes

1. Other connectionist approaches to linguistics appealing to the notion of harmony include John Goldsmith's "harmonic phonology" (Goldsmith, to appear) and George Lakoff's "cognitive phonology" (Lakoff, 1988).

2. In addition, there are corresponding problems related to learning, but these are not yet relevant to this research.

3. In Smolensky (1986b), the dynamical question is also considered: do the lower and higher level models evolve isomorphically in time? In this paper, we do an end run around the dynamics, working directly with the optimal equilibrium states the (incompletely specified) dynamics is trying to find.

4. Smolensky (in press 2) uses this technique as the technical basis for a reply to the putative dilemma of Fodor & Pylyshyn (1988): connectionism must choose between associationist and structure-sensitive processing.

5. This simple form for H arises from treating biases as weights to an extra unit with constant value 1, and treating input lines as though they originated in units interior to the network. In this form, H in the text is maximized when each unit achieves its maximum or minimum activation values. Networks whose units are not driven to their limits — e.g., quasi-linear units with sigmoid non-linearities, discussed in Smolensky (1986b) and very popular since Rumelhart, Hinton, & Williams (1986) in "back-propagation networks" — can be analyzed by adding to H a term $-\sum_i h(a_i)$ which does not introduce further interactions among the units, but is designed to penalize units with extreme values. E.g., for the popular logistic non-linearity $a_i = (1+e^{-input_i})^{-1}$, we set $h(a) = a \ln a + (1-a)\ln(1-a)$. This function h, like the other terms in H, has an interpretation in terms of statistical inference and information theory.

6. This constraint is particularly important since without it, every pair of context and individual predicate would have its own free parameter, giving rise to 572 parameters of this type alone — with only 760 data points to fix the parameters.

References

Cohen, M. A. & Grossberg, S. (1983). *Absolute stability of global pattern formation and parallel memory storage by competitive neural networks.* IEEE Trans SMC-13, 815–825.

Fodor, J. A. & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition,* **28**, 3–71.

Golden, R. M. (1986) The "Brain-State-in-a-Box" Neural Model Is a Gradient Descent Algorithm. *Mathematical Psychology,* **30-1**, 73–80.

Golden, R. M. (1988) A Unified Framework for Connectionist Systems. *Biological Cybernetics,* **59**, 109–120.

Goldsmith, J. (to appear). Harmonic phonology. To appear in J. Goldsmith (ed.) *Proceedings of the Berkeley Workshop on Nonderivational Phonology.* Univ. of Chicago Press.

Hinton, G.E. & Sejnowski, T.J. (1983). Analyzing cooperative computation. *Proceedings of the Fifth Annual Conference of the Cognitive Science Society.* Rochester, NY.

Hinton, G.E. & Sejnowski, T.J. (1986). Learning and Relearning in Boltzmann Machines. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations.* Cambridge, MA: MIT Press/Bradford Books.

Hopfield, J.J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences, USA* **79**, 2554–2558.

Hopfield, J.J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences, USA* **81**, 3088–3092.

Hopfield, J.J. (1987). Learning algorithms and probability distributions in feed-forward and feed-back networks. *Proceedings of the National Academy of Sciences, USA* **84**, 8429–8433.

Lakoff, G. (1988). A suggestion for a linguistics with connectionist foundations. In D. Touretzky, G. Hinton, & T. Sejnowski (eds.), *Proceedings of the 1988 Connectionist Models Summer School.* Morgan Kaufmann Publishers.

Legendre, G., Miyata, Y. & Smolensky, P. (1990) *Harmonic Grammar - A formal multi-level connectionist theory of linguistic well-formedness: An application.* Technical report #90-4, Institute of Cognitive Science, University of Colorado at Boulder.

Rumelhart, D.E., Hinton, G.E., & Williams, R.J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations.* Cambridge, MA: MIT Press/Bradford Books.

Smolensky, P. (1983). Schema selection and stochastic inference in modular environments. *Proceedings of the National Conference on Artificial Intelligence.* Washington, DC.

Smolensky, P. (1986a). Information processing in dynamical systems: Foundations of harmony theory. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations.* Cambridge, MA: MIT Press/Bradford Books.

Smolensky, P. (1986b). Neural and conceptual interpretations of parallel distributed processing models. In J. L. McClelland, D. E. Rumelhart, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 2: Psychological and biological models.* Cambridge, MA: MIT Press/Bradford Books.

Smolensky, P. (1988). On the proper treatment of connectionism. *The Behavioral and Brain Sciences.* **11**, 1–23.

Smolensky, P. (in press 1). Tensor product variable binding and the representation of symbolic structures in connectionist networks. *Artificial Intelligence.*

Smolensky, P. (in press 2). Connectionism, constituency, and the language of thought. In B. Loewer & G. Rey (Eds.), *Fodor and his critics.* Blackwell's.

A Parallel Constraint Satisfaction and Spreading Activation Model for Resolving Syntactic Ambiguity

Suzanne Stevenson
Department of Computer Science
University of Maryland, College Park

Abstract

This paper describes a computational architecture whose emergent properties yield an explanatory theory of human structural disambiguation in syntactic processing. Linguistic and computational factors conspire to dictate a particular integration of symbolic and connectionist approaches, producing a principled cognitive model of the processing of structural ambiguities. The model is a hybrid massively parallel architecture, using symbolic features and constraints to encode structural alternatives, and numeric spreading activation to capture structural preferences. The model provides a unifying explanation of a range of serial and parallel behaviors observed in the processing of structural alternatives. Furthermore, the inherent properties of active symbolic and numeric information correspond to general cognitive mechanisms which subsume a number of proposed structural preference strategies.

1 Introduction

It has been repeatedly demonstrated that people have little trouble in processing structurally ambiguous sentences; moreover, they yield consistent structural preferences in the face of ambiguity. Yet theories of human structural preferences have progressed little beyond the stage of unrelated descriptions of each piece of the psychological data. The research described here aims to shed light on the cognitive principles used in structural disambiguation by exploring the computational mechanisms which underlie them. The goal is not to create a parser in which human structural preferences are built in, but to design a parsing architecture whose basic properties *predict* those preferences.

A predictive theory of structural disambiguation emerges from the active, distributed nature of the computational model described here. The model is a hybrid massively parallel architecture, combining symbolic constraint satisfaction and numeric spreading activation. Symbolic features and constraints based on Chomsky's (1981, 1986) Government-Binding theory (GB) capture multiple structural alternatives in parallel in a linguistically motivated way. Numeric spreading activation guides the choice between these structural alternatives by encoding and integrating the degree of featural compatibility, the recency of activation, and the strength of lexical preference for each possible attachment. The preference for a particular structural attachment is thus uniformly determined by an inherent mechanism of the architecture.

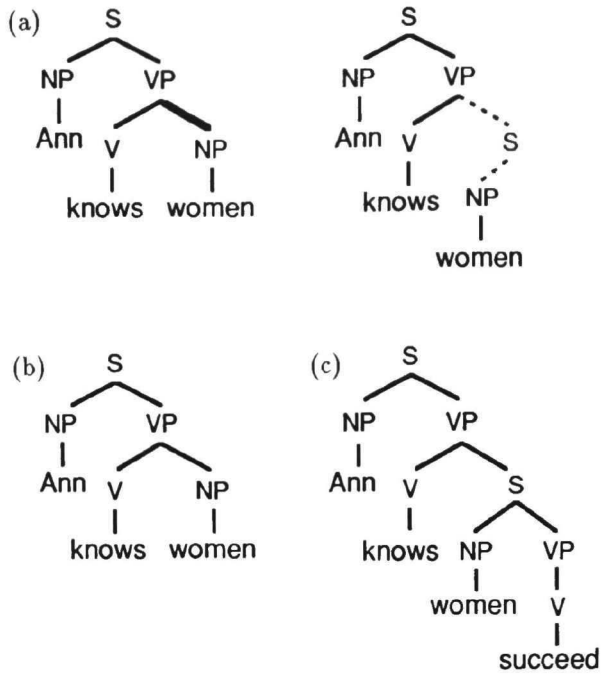
The remainder of this paper discusses the model and its consequences in more detail. Section 2 describes the key psycholinguistic issues which must be addressed by an explanatory theory of structural disambiguation. Section 3 presents the details of the hybrid architecture and describes the properties from which its structural disambiguation behavior emerges. In Section 4, the explanatory power of the mechanism is demonstrated. Section 5 discusses some related work on structural disambiguation, and Section 6 concludes the paper with a summary of the contributions of this research.

2 The Psycholinguistic Data

A structural ambiguity gives rise to multiple attachment possibilities for a syntactic phrase. Any model of structural disambiguation must address the two issues of how to process the valid structural alternatives for the phrase, and how to capture the structural preference factors which choose between them. This section describes some of the psycholinguistic observations related to these issues.

Serialism versus parallelism

One of the first issues that must be addressed in building a predictive model of structural preferences is the degree to which structural alternatives are processed in parallel. This is, in fact, a major open question in psycholinguistic research. Clear evidence for a serial mechanism comes from experiments which demonstrate consistent strong preferences for one resolution over another of a temporary ambiguity (Frazier, 1978). For example, in the sentence beginning:



(a) ——— preferred ------ non-preferred

(b) preferred resolution of the ambiguity.

(c) non-preferred resolution of the ambiguity.

Figure 1: Attachment possibilities for *women*.

Ann knows [NP women] ...,

the attachment of the NP is temporarily ambiguous, as shown in Figure 1. The sentence may end after *photo* or continue "...succeed," each case resolving the ambiguity in a different way. The consistent preference for the first of these resolutions of the NP attachment indicates that the parser chooses one of the possible structural alternatives and pursues it serially.

However, equally convincing evidence for a parallel architecture emerges from experiments which reveal the availability of multiple structures in on-line processing of these temporary ambiguities (Gorrell, 1987). That is, experiments indicate that in the above example, both the NP–to–VP and the NP–to–S attachment possibilities are maintained in parallel. A major contribution of the model proposed here is that it naturally accounts for these apparently contradictory results, while other models have failed to do so. Exploiting the interesting interaction of serial and parallel qualities in fact leads to the model's ability to provide an architectural explanation for the range of serial and parallel behaviors attested in a number of psycholinguistic experiments. Section 3 describes the hybrid model which turns this tension between serial and parallel processing to its advantage.

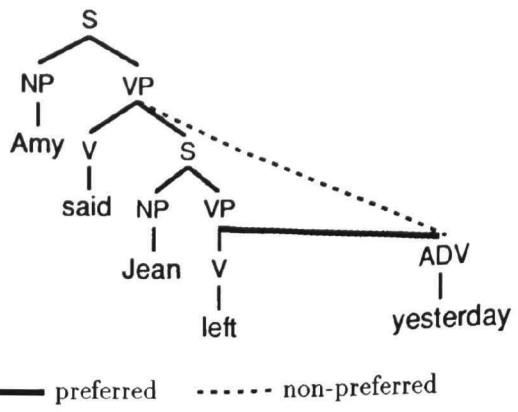


Figure 2: Attachment possibilities for *yesterday*.

Structural preference accounts

Numerous principles have been proposed to account for the consistent structural preferences displayed by the human parser in the face of syntactic ambiguity. For example, Minimal Attachment asserts that when the parser has a choice between two or more ways of attaching the current phrase into the parse tree, it will pick the one which requires the creation of the fewest number of new nodes (Frazier, 1978). The preference in Figure 1 for the NP to attach as the object of the verb is a clear case of Minimal Attachment: the parser prefers to attach the NP directly to the VP, rather than creating the S node to serve as its attachment site.

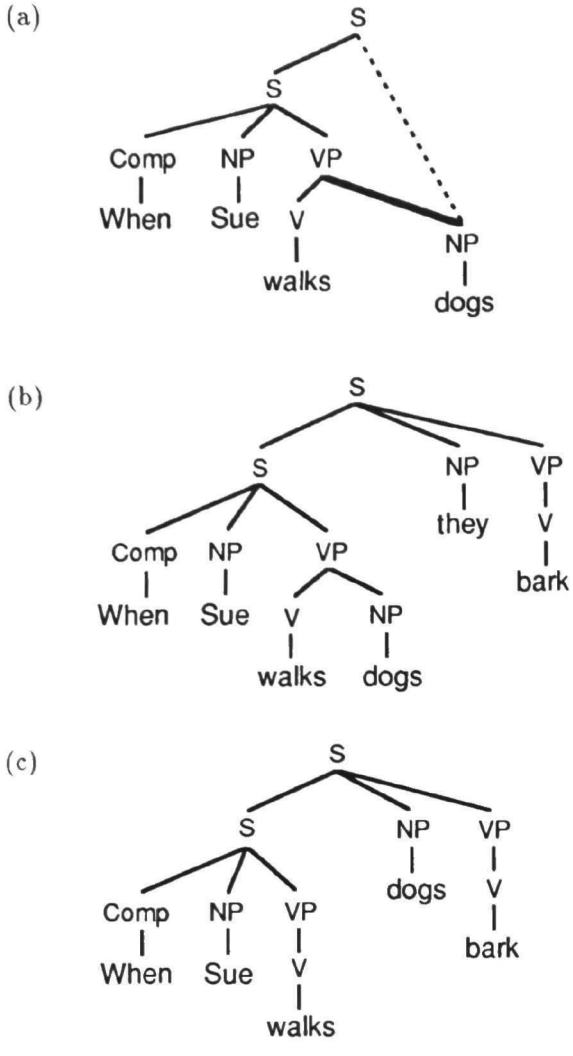
Another principle, Late Closure, states that the parser will prefer to keep a constituent open (that is, available to attach into) as long as possible, entailing that people will prefer to attach a phrase into the most recent open constituent (Frazier & Rayner, 1982). Late Closure accounts for the preferences indicated in Figures 2 and 3. In each case, the non-preferred attachment would require first closing off the open subordinate verb phrase.

Theories of lexical preferences claim that verbs have varying strengths of expectation for their possible arguments. For example, in these sentences taken from Ford, Bresnan, & Kaplan (1982):

Joe included [NPthe package [PPfor Susan]].
Joe carried [NPthe package] [PPfor Susan].

the PP attachment preferences indicated by the given bracketings result from a difference in how strongly the verbs *include* and *carry* expect a PP argument.

These are just a few of the many such accounts of the factors involved in structural disambiguation (for example, Kimball, 1973; Frazier & Fodor, 1978; Nicol, 1988; McRoy & Hirst, 1989). Although valuable as concise descriptions of a wealth of psycholinguistic phenomena, these accounts do not explain *why* the parser



(a) ——— preferred ------ non-preferred

(b) preferred resolution of the ambiguity.

(c) non-preferred resolution of the ambiguity.

Figure 3: Attachment possibilities for *dogs*.

follows these particular patterns of structural preferences and not others. On closer examination, many of these psycholinguistic principles can be shown to be specific statements of the results of more general cognitive processes: the impetus to immediately structure incoming material, the decrease in salience of a structure over time, and the increase in salience given higher frequency or priming. In the model described here, these processes are precisely mirrored in its inherent qualities of active distributed computation, decay of activation, frequency-encoding weights, and activation of expectations. The properties of its active distributed computation are in turn strongly influenced by the model's integration of serial and parallel processing. Section 4 presents in detail how these emergent properties of the architecture predict the observed pattern of preferences.

3 The Hybrid Architecture

The rise over the past decade of the connectionist approach to cognitive modeling has generated much debate over the relative merits of serial symbolic processing models and massively parallel architectures restricted to numeric spreading activation. The debate has sparked an interest in so-called "hybrid" models which attempt to exploit the desired properties of each approach, while avoiding their respective pitfalls. A precise formulation of the components of the structural disambiguation process has motivated the design of a hybrid model in the research presented here.

Structural disambiguation involves two distinct parsing processes: identifying the allowable attachments for a phrase, and choosing between them. The first process involves the linguistic *competence* of the parser; it uses grammatical knowledge to select the valid structural alternatives. The second process is a matter of linguistic *performance*; extragrammatical factors are taken into account in determining the structural preferences. The motivation for a hybrid approach to a computational explanation of structural disambiguation arises from the necessity of capturing within a single model the properties relevant to both of these processes. Traditional serial symbolic processing models have been good at encoding discrete competence knowledge, while connectionist models are quite successful at integrating the multitude of factors affecting performance.

The question, of course, is how to combine these divergent approaches in a principled way. This can be achieved more naturally than might be expected. Properties of competence and performance themselves each converge on some type of massively parallel architecture. On the competence side, a recent trend in linguistic theory has been away from unwieldy, construction-specific rule-based systems toward a so-called "principles and parameters" approach. Government-Binding theory (GB), founded on this approach, is a constraint-based theory in which the validity of syntactic structures is determined by local licensing relations among constituent phrases. An active, distributed architecture lends itself well to the formulation of grammatical knowledge as a set of simultaneous declarative constraints which must be satisfied locally.

On the performance side of the issue, processing structural attachments requires some interesting interaction of serialism and parallelism, as noted in Section 2. Spreading activation through a parallel network inherently combines aspects of serial and parallel processing. Although highly parallel in its simultaneous communication to all neighboring nodes, activation is intrinsically serial in its spread through the space of the network. Furthermore, the massive parallelism of activation must be harnessed through some kind of focusing mechanism, or its sole effect will be network saturation.

Thus the problem domain itself strongly supports a massively parallel architecture combining symbolic constraint satisfaction and numeric spreading activation.

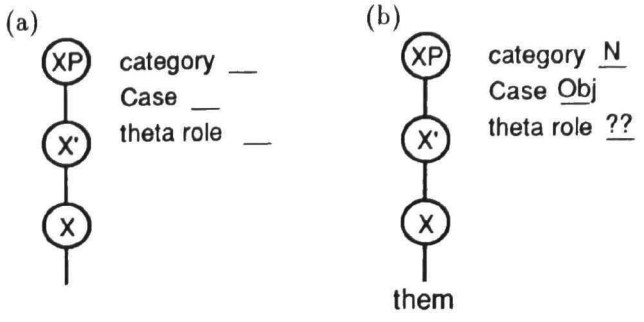
The design of a hybrid architecture must address the issue of how to integrate the seemingly incompatible properties of the symbolic and connectionist processing paradigms. As shown in Table 1, traditional symbolic models typically manipulate symbols serially, building new structure to solve a problem. Connectionist models, on the other hand, compute numeric activation functions in parallel, solving problems by activating the appropriate built-in structures. In the model described here, linguistic and computational principles have converged on a profitable synthesis of these approaches along each of the three relevant dimensions.

First consider the units of information. In the model, symbolic constraint satisfaction naturally encodes GB, while numeric spreading activation acts as a uniform mechanism to capture diverse sources of structural preference information. Many possibilities have previously been explored for effectively integrating symbolic and numeric computation in a cognitive model: Symbolic and numeric computation may operate at different levels of abstraction (Hendler, 1987); they may operate at the same level of abstraction, but independently (Waltz & Pollack, 1985); and activation may constrain the passing of symbolic information (Hendler, 1986). The model here incorporates a new approach, in which symbolic features constrain the spreading of activation. Symbolic constraint satisfaction directly affects the numeric activation of a node, and determines the paths along which activation can spread beyond the node. This is accomplished by using an activation function which depends in part on the *state* of a node, which is a numeric estimation of its degree of constraint satisfaction.

Both linguistic and computational reasons motivate this technique. Symbolic features, which represent linguistic competence, control what is affected by numeric activation, which guides performance. The primacy of symbolic information has a positive computational effect, since it restrains the unwanted spreading of activation. For example, many nodes represent potential syntactic attachments which will be determined to not satisfy the necessary grammatical constraints. If activation could spread across these bad attachments, lending

Symbolic Processing	Connectionism
symbols	numeric activation
serialism	parallelism
dynamic structure (creation)	fixed structure (recognition)

Table 1: Properties of Symbolic Processing and Connectionist Models.



(a) $\overline{X}$ template and sample features.

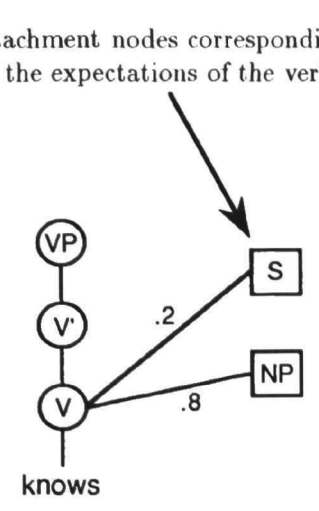
(b) template instantiated and initialized by *them*.

Figure 4: The parser's generic syntactic phrases.

other nodes false support, it would make the determination of structural preferences based on activation much more difficult. Instead, a negative *state*, indicating invalid feature values, forces a node's activation to zero even when it is receiving external stimuli.

Second, the issue of dynamic versus fixed structure must be resolved. Most massively parallel parsers are based on a fixed network of nodes (Cottrell, 1985; Selman & Hirst, 1985); only the model of Waltz & Pollack (1985) has made use of dynamic structure creation. However, their model uses a traditional serial structure-building parser to construct a network corresponding to the parse(s) of the input. The model here strikes a compromise between a totally fixed network structure and the ability to create an arbitrary network structure on the fly. The parser is limited to a single fixed phrase structure template, but it may instantiate this template at will and connect the instances to the input. This "generic" syntactic phrase is then passed initializing features by the associated input. Figure 4 shows the template and a sample instantiation. All logical possibilities of inter-phrase attachments are represented by dynamically allocated attachment nodes; constraint satisfaction rules out those attachments that are invalid.

Once more, this approach is motivated by both linguistic and computational factors. The generic phrasal template is inspired by the lack of general phrase structure rules in GB. $\overline{X}$ theory, a subsystem of GB, conceives of all phrases as having the same fixed structural shape, with differences in grammatical behavior entailed by features projected from the input. From a computational perspective, this vastly simplifies the structure-building component by restricting it to uniform instantiation of fixed templates. The approach also incorporates a lexically-driven aspect which allows the model to respond to conditions in the input in a straightforward way. For example, a verb can easily determine the weights on its connections to attachment nodes, dynamically taking into account the frequency



The verb sets the weights on the connections to nodes which represent potential attachments.

Figure 5: Weights capture lexical frequencies.

of its various arguments, as shown in Figure 5.

Finally, the model incorporates elements of both serial and parallel computation. In other massively parallel parsers, the serial aspect of spreading activation is the only constraint on the parallelism of the computation. Here the parallelism is further restricted, by prohibiting top-down precomputation of phrase structure. An $\overline{X}$ node may trigger multiple attachment alternatives, but it cannot cause instantiation of phrase structure based solely on an expectation. For example, in Figure 5 the verb may establish attachment nodes corresponding to its NP and S expectations, but it cannot license the building of an $\overline{X}$ phrase for either of these alternatives. Although quite restricted, the model is still highly parallel in that alternative attachments based on expectations and on phrases built by bottom-up evidence exist in parallel and compete for activation.

Again, both linguistic and computational concerns support this approach. According to GB, phrase structure is projected from a lexical item; no $\overline{X}$ phrase can exist without being licensed in this way. Interpreting this principle computationally as a constraint on structure building, rather than one which checks already computed structure, increases the efficiency of the approach. Disallowing precomputation in the model not only limits the number of nodes that are created, but also simplifies the structure building, constraint satisfaction, and spreading activation algorithms significantly.

Not only do each of these decisions receive independent linguistic and computational support, these issues are in fact a set of interrelated choices. The motivation for integrating symbolic and numeric computation was presented above. By granting symbolic information a primary role in the processing of the network, the opportunity for building structure arises. Dynamically creating structure in turn discourages top-down precomputation. The fact that these are mutually constraining decisions yields a hybrid architecture with a coherent cluster of properties. The following section demonstrates that the combined result of these interdependent choices is a principled architecture from which the desired structural disambiguation behavior emerges.

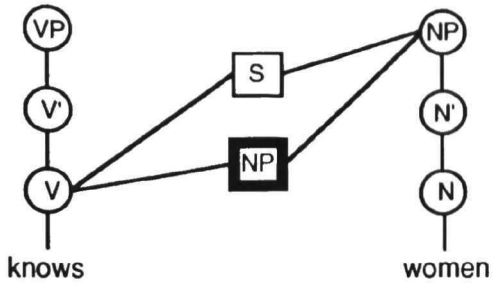
4 Predictions of the Model

Section 2 presented the two types of psycholinguistic observation which a theory of structural disambiguation should explain: the conflicting evidence concerning the degree to which structural alternatives are maintained in parallel, and the pattern of structural preferences which people exhibit. This section describes the behavior of the model relevant to these two issues. First, the critical properties which underlie the model's behavior will be presented. Next, the model's restricted parallelism will be seen to resolve the seemingly contradictory evidence for serial and parallel processing of structural alternatives. Finally, the preference behavior of the model on some illustrative cases of structural ambiguity will be discussed.

The property of the model which yields a unified theory of structural disambiguation is the process of active communication of symbolic and numeric information within the parsing network. When a phrase is created, the parser establishes nodes for the potential attachments of the phrase into the parse tree. The phrase must then actively communicate features to its neighbors in order to determine which attachments are valid. These attachment nodes are the structural alternatives for the phrase; their activation level encodes their relative preference. They receive numeric activation from their neighbors across weighted connections; these weights encode the frequency with which one phrase expects to attach to another. Activation decays over time if it is not reinforced. A competitive activation method (Reggia, Marsland, & Berndt, 1988) provides a focusing mechanism to sharpen the preference for an alternative.

Integrating serial and parallel behavior

Parallelism is restricted in the model by disallowing top-down precomputation. Thus a phrase which is actively seeking an attachment can only communicate with other structures which have received evidence from the input. In addition, a competitive mechanism required to control spreading activation attempts to focus activation on a single alternative. These two properties lead to observed serial behavior in the parser, accounting for the results of psycholinguistic experiments which support a serial mechanism. For example, the



The NP may attach as the S object of *know* or as the NP object of *know*. Since the features of the first attachment expectation are incompatible with the NP, only the second attachment node remains active.

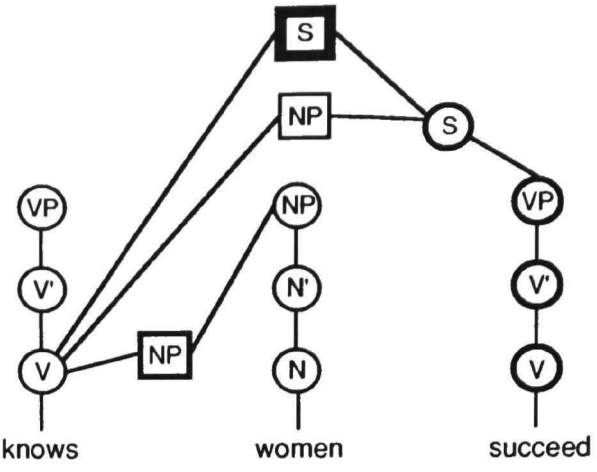
Figure 6: Initial attachment possibilities for *women*.

consistent preference for the NP–to–VP attachment in Figure 1 is simply due to the fact that it is the *only* attachment initially available for the NP. The parser has not encountered any overt evidence of a sentential object, so the NP can only make the attachment directly to the VP, as shown in Figure 6. In cases where multiple attachments are available, as was the case in Figure 2, the competitive activation mechanism will focus on one of them, also leading to seemingly serial behavior.[1] In either case, evidence that the initially preferred attachment is incorrect leads to a delay in processing, since the new structural alternative must compete for activation with the established attachment. This serial behavior mimics that of the human parser demonstrated in the analysis of eye-movements recorded while people read these types of temporarily ambiguous sentences (Frazier & Rayner, 1982).

This behavior is not inconsistent with a fundamentally parallel architecture, however. The model maintains multiple structural alternatives for which evidence exists, and projects active expectations. These expectations are particularly relevant in accounting for data showing that multiple alternatives are in some form available to the parser even when evidence from the input has not been encountered. Gorrell (1987), using a lexical decision task immediately following the NP, showed that the non-preferred structural alternative in Figure 1 (that is, the NP–to–S attachment) could prime a verb. Gorrell took this as clear evidence of precomputation of all possible structural alternatives. However, the active expectations of the model here can easily account for the observed behavior. Figure 7 shows that since the verb *know* actively expects either an NP or S argument, the S expectation primes the subsequent occurrence of a verb in the input.[2]

[1] The factors involved in focusing preferences will be discussed below.

[2] The verb in English carries tense features which trigger the building of an S node. More precisely, it is the attachment of this S node to the VP which is primed by the S expectation.



The verb's active attachment nodes encode the expectation of an NP or S object. The S expectation primes the S projected from the tense features of the verb.

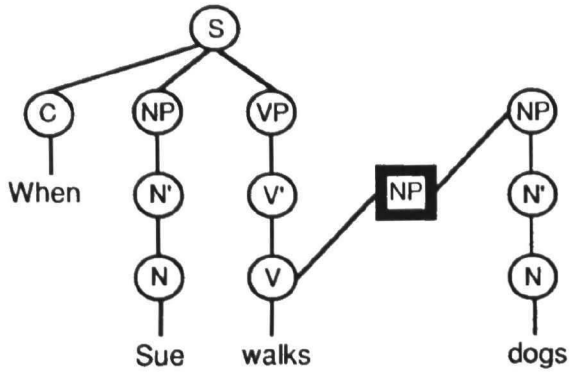
Figure 7: Priming of a subsequent S.

Thus, even with the restriction of only maintaining the knowledge of alternative possibilities, rather than precomputing their structure, the model predicts the results of experiments supporting parallelism in situations of structural ambiguity. On the other hand, the restrictedness of the parallelism accounts for the serial effects displayed in experiments testing structural preferences. This explanation of the range of behaviors in processing structural ambiguities is the result of the inherent integration of aspects of serialism and parallelism in the model.

Unifying structural preferences

Fundamental properties of the model directly relate to the cognitive principles responsible for human structural preferences that were noted in Section 2. The need for a phrase to actively determine its valid attachments predicts that people will show a preference for attachments which allow immediate structuring of input. The decay of activation explains recency effects in structural preference, which correspond to a decrease in salience of older attachment sites. Weights increase salience by strengthening activation of more frequent alternatives. And finally, activation of expectations primes certain alternatives. These structural preference mechanisms interact in the resolution of each instance of structural ambiguity. A predictive model of structural disambiguation thus arises from these natural properties.

For example, the active attachment behavior of the model, in conjunction with the restricted parallelism discussed above, provides a simple account of many of the Minimal Attachment and Late Closure cases. Fig-



Only one attachment is initially available for the NP: as the NP object of *read*. (No main clause S node has been projected.)

Figure 8: Initial attachment possibilities for *dogs*.

ures 6 and 8 demonstrate that the alternative attachments are just not available at the time the NP is processed. Thus Minimal Attachment, represented in Figure 6, results not from an explicit comparison of the complexity of various choices as in previous models (for example, Frazier, 1987; Gorrell, 1987; Clark, 1988), but from the active nature of the model's syntactic phrases. Furthermore, cases of Late Closure as in Figures 8 are accounted for by the same properties of the model. That is, when the NP begins to actively seek an attachment, the parser has not yet projected a main clause S node, so the embedded VP is the only possible attachment site. This gives a uniform explanation of a range of structural preferences for which two attachment strategies were previously thought to be required.

The other standard cases of Late Closure, which were exemplified in Figure 2, are predicted by the recency effects which result from the decay of activation. Although both attachments are available, the higher attachment has less activation because that verb is more distant in the input and its activation has decayed. As the two verbs actively compete for the adverbial attachment, the more recent verb has the advantage of more activation and will "win" the competition. A similar effect arises from lexical frequencies, which are encoded by the weights on connections to possible arguments of a verb, and provide more or less advantage to potential attachments. (See Figure 5.) Priming in the form of an expectation also leads to the active advantage of an attachment, as discussed above and demonstrated in Figure 7.

Thus we have a model which accounts for a wide range of structural preferences with the single principled mechanism of active symbolic and numeric information. Not only does the model predict the various preferences, but it does so with a spreading activation mechanism which naturally integrates their interaction as well.

5 Related Approaches

Recent work of McRoy & Hirst (1989) similarly attempts to unify a broad range of syntactic influences on structural preferences. However, the timing effects they seek are not a natural result of their parsing architecture; in fact, they must explicitly build in the interaction of preferences and timing. Furthermore, their serial, race-based parser is unable to account for the parallel aspects of the processing of ambiguities, as is the similar model of Frazier & Rayner (1982). Cottrell's (1985) connectionist parser results in Minimal Attachment behavior due to the nature of spreading activation, but falls short of accounting for the related principle of Late Closure with the same mechanism. In the parallel models of Gorrell (1987) and Clark (1988), the ranking mechanisms proposed for determining structural preferences are not a fundamental aspect of the parsing architecture, and each fails to capture recency and lexical preference effects.

6 Conclusions

This paper has presented a hybrid massively parallel parsing architecture which integrates symbolic and numeric processing in a linguistically and computationally motivated way. Behavior mimicking human processing of structural ambiguities emerges from the inherent properties of this architecture. The serial aspects of spreading activation and the restriction on top-down precomputation provide a natural explanation for the seemingly irreconcilable range of serial and parallel behaviors in processing structural alternatives. The property of active symbolic and numeric information leads to a principled account of structural preferences, unifying with a single mechanism the effects of various previously proposed preference strategies, such as Minimal Attachment and Late Closure.

Acknowledgments

This work has been supported by University of Maryland Graduate School Fellowships, NSF Grant IST-8451430, and the University of Maryland Institute for Advanced Computing Studies (UMIACS).

Thanks to Amy Weinberg, James Reggia, and Sven Dickinson for helpful comments on earlier drafts of this paper.

References

Chomsky, N. (1981). *Lectures on Government and Binding: The Pisa Lectures.* Dordrecht: Foris Publications.

Chomsky, N. (1986). *Barriers.* Cambridge: MIT Press.

Clark, R. (1988). "Parallel Processing and Local Optimization." Talk given at the University of Maryland Processing Workshop, December 9, 1988.

Cottrell, G. W. (1985). "Connectionist Parsing." *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*, 201–211.

Ford, M., J. Bresnan, and R. Kaplan (1982). "A competence-based theory of syntactic closure." In J. Bresnan (Ed.), *The Mental Representation of Grammatical Relations.* Cambridge: MIT Press.

Frazier, L. (1978). *On Comprehending Sentences: Syntactic Parsing Strategies.* Unpublished doctoral dissertation, University of Connecticut. Distributed by Indiana University Linguistics Club.

Frazier, L. (1987). "Sentence processing." In M. Coltheart (Ed.), *Attention and Performance XII.* Hillsdale, NJ: LEA.

Frazier, L., and J. D. Fodor (1978). "The Sausage Machine: A new two-stage parsing model." *Cognition* 6, 291–325.

Frazier, L., and K. Rayner (1982). "Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences." *Cognitive Psychology* **14**, 178–210.

Gorrell, P. (1987). "Structural ambiguity and syntactic priming: Toward a theory of ranked-parallel parsing." Manuscript, University of Maryland.

Hendler, J. (1986). Integrating Marker-Passing and Problem Solving: A Spreading Activation Approach to Improved Choice in Planning." Technical Report 1624, Computer Science Department, University of Maryland, College Park, Maryland.

Hendler, J. (1987). "Marker-passing and Microfeatures." *Proceedings of the Tenth International Joint Conferences on Artificial Intelligence*, 151–154.

Kimball, J. (1973). "Seven principles of surface structure parsing in natural language." *Cognition* **2**:**1**, 15–47.

McRoy, S. and G. Hirst (1989). "Race-Based Parsing and Syntactic Disambiguation." Manuscript, University of Toronto. Submitted for publication.

Nicol, J. (1988). "Coreference Processing During Sentence Comprehension: A Review of On-Line Research." Manuscript, University of Arizona.

Reggia, J., P. Marsland, and R. Berndt (1988). "Competitive Dynamics in a Dual-Route Connectionist Model of Print-to-Sound Transformation." *Complex Systems.*

Selman, G., and G. Hirst (1985). "A Rule-Based Connectionist Parsing Scheme." *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*, 212–219.

Waltz, D., and J. Pollack (1985). "Massively parallel parsing: A strongly interactive model of natural language interpretation." *Cognitive Science* **9**, 51–74.

FUNCTIONAL CONSTRAINTS ON BACKWARDS PRONOMINAL REFERENCE

Catherine L. Harris
Elizabeth A. Bates

Department of Cognitive Science
University of California, San Diego

ABSTRACT

How does the syntax of a sentence constrain speakers' selection of pronominal referents? Drawing on work by functionalist grammarians, we describe the communicative effect of using a pronoun vs. a definite noun phrase, a matrix vs. a subordinate clause, and the simple past tense vs. anterior/imperfective aspect. Our analysis allowed us to predict differences in coreference judgements for the following three sentence types:

He worked on a top-secret project when John was ordered to quit.
He was working on a top-secret project when John was ordered to quit.
When he worked on a top-secret project, John was ordered to quit.

Coreference judgements from 70 speakers supported our predictions and our research program: An adequate characterization of how syntax constrains sentence comprehension requires reference to the communicative functions performed by syntactic forms.

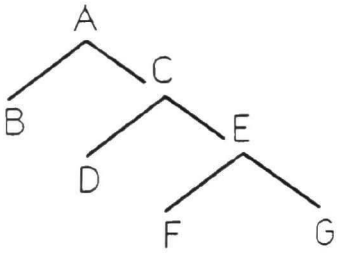
INTRODUCTION

Pronouns are typically used to refer to persons or objects previously identified by name in a sentence or discourse, as illustrated by (1). In some sentences, pronouns can precede their referents, as in (2), while in others, such as (3), native speaker intuitions are that coreference between an initial pronoun and a subsequent lexicalized noun is blocked.

(1) Harry met Sally when he moved to New York.
(2) When he moved to New York, Harry met Sally.
(3) He moved to New York when Harry met Sally.

Because the only change in form across these examples is syntactic, syntax must be responsible for differences in perceived meaning and differences in coreference judgements. How should these changes in syntactic form be characterized, and how do these forms aid listeners in determining the referent intended by speakers?

Linguists have long noted that the relationship between nodes in a phrase-structure tree appears to constrain coreference assignment. Reinhart (1983) has argued that coreference is blocked (prohibited) when an anaphoric element **c-commands** a lexical noun. If the anaphoric element does not c-command the candidate referent, then coreference is free to vary with semantic and pragmatic factors.



A relatively accessible definition of c-command is provided by Radford (1988): A node c-commands its sisters (those nodes on the same level) and their descendants. In the tree at the right, node D c-commands E, F, G, but not A, B, or C. In example (3), *he* is the sentence subject, and thus c-commands *Harry* as well as all other sentence elements. In (2), however, *Harry* c-commands *he*.

Numerous linguistic analyses have questioned whether c-command makes the right predictions in all cases where it is applicable (Bolinger, 1979; Bosch, 1984; Kuno, 1987). For example, because *he* is the sentence subject in (4) and (5), it c-commands *John*. The prediction is thus that coreference should be blocked in both of these sentences. Linguistic intuition is, however, that an interpretation of coreference is odd in (4) but reasonably felicitous in (5). (Examples are from Carden, 1980, cited in Kuno, 1987.)

(4) Near John, he found a snake.
(5) Near the girl John was talking with, he found a snake.

From a functionalist perspective, even if c-command (or some other statement of a surface relation between sentence elements) did perfectly divide all utterances into categories such as blocked and free, the job of understanding how syntactic form constrains meaning would be incomplete. What would remain to be developed is an explanation of why this syntactic device (eg., level and direction of embedding) predicts speaker coreference judgements. If the surface devices that comprise c-command prove to be only partially predictive of coreference judgements, we would still want to determine the communicative reasons why this partial correlation between form and meaning is a useful one for selecting pronominal referents.

GENERALIZATIONS ABOUT SOME FORM-FUNCTION MAPPINGS

The sentences in (6)-(9) differ in the grammatical devices they use, but they contain almost identical lexical items and describe roughly the same events. Next to each sentence we have noted our intuitions about whether coreference between the proper name and pronoun is possible.

(6) **likely:** Bill stayed with some underground writers when he was ordered to leave the country.
(7) **optional:** When he stayed with some underground writers, Bill was ordered to leave the country.
(8) **optional:** He was staying with some underground writers when Bill was ordered to leave the country.
(9) **blocked:** He stayed with some underground writers when Bill was ordered to leave the country.

The obvious changes in form between these sentences are the differences between lexicalized noun (*Bill*) and pronoun, the difference between subordinate and matrix clause, and the difference between imperfective aspect (*was staying*) and the completive aspect which is typically indicated in English with the simple past tense (*stayed*).

In the following sections, we sketch a functionalist explanation of how these differences in syntactic forms lead to differences in coreference judgements. We will then investigate the extent to which speaker judgements of coreference support our hypotheses about form-function mappings.

PRONOUNS AND DEFINITE NOUN PHRASES

What factors influence whether a speaker will use a pronoun or a full nominal to indicate reference to a person or object? Givon (1984) and Prince (1981) have described how pronouns are typically used when the intended referent is the discourse topic, or is otherwise recoverable from context. Use of a pronoun thus tends to be a signal to the listener that the current topic is unchanged: "I'm still talking about the same thing." In contrast, a speaker uses a full lexical noun to indicate a change in topic or when the identity of the pronoun can not be recovered from context. Even if a listener could, with some effort, infer the identity of a referent from the context, a lexical noun may still be used if substantial time has passed since last mention of the entity, or if other discourse characters have been discussed in the interim since last mention of the intended referent.

In the likely assignment example in (6), coding *Bill* as the sentence subject establishes him as the discourse topic. It is thus most natural to interpret the following *he* as referring to *Bill*. In the blocked

case (9), a pronoun (*he*) is the sentence subject and thus established as the current discourse topic.[1] The lexicalization of *Bill* in the following clause signals that *Bill* is new information, and that either a shift in topic has occurred or that the speaker wishes to refer to a discourse entity other than the current topic. The resulting inference is that *he* has an extrasentential reference, and that Bill and *he* refer to different people.

This helps explain the difference in referent selection between the **blocked** and **likely** assignment cases, but does not shed light on why backwards pronominalization is allowed when the pronoun appears in a subordinate clause. The next section addresses this point.

MATRIX AND SUBORDINATE CLAUSES

Clauses conjoined with conjunctives/adverbials such as *when, because, before, after, while* are typically viewed by syntacticians to be matrix-subordinate pairs: the clause introduced by the adverbial is the subordinate clause, and the clause beginning with a head noun is the matrix clause. For the purposes of the current paper we will accept this definition of matrix and subordinate.

What is the communicative function of conjoining two events in a matrix-subordinate pair? Matthiessen and Thompson (1988) note that crosslinguistically, matrix-subordinate clause combining involves some type of circumstantial relation: the subordinate clause signals a condition, reason, purpose, cause, setting, manner, or means. The matrix clause is understood to be the main or focal assertion. The subordinate clause thus functions as the *context* in which to interpret or evaluate the speaker's main assertion.

A similar conception of the role of subordinate clauses has been advanced by Chafe (1988), who views matrix-subordinate clause combining as a grammaticization of what he calls *event linking*. He notes that when the subordinate clause is placed before the matrix, the clause linkage is anticipatory: the first clause must be followed by a second clause (barring a change of mind on the part of the speaker). When the matrix clause appears first, however, the listener may not know until the second clause is actually encountered that the utterance is an instance of event linking.

Why might **listeners** be willing to let a pronoun in a sentence-initial *subordinate* clause -- but not a pronoun in a sentence-initial *matrix* clause -- be coreferential with a lexical noun in a subsequent clause? Because sentence-initial subordinate clauses require a following clause, listeners know that material relevant to the current clause is immediately forthcoming. Because subordinate clauses function as a context in which to evaluate or interpret a main point, listeners will attempt, where possible, to integrate material across the clauses, or to seek reasons why material in the first is relevant to the second.

Why would **speakers** place a pronoun before a referent, violating the convention that a pronoun signals continued reference to an established topic? We doubt that the speaker's *goal* is to violate this convention. Instead, violation should be viewed as a side-effect of other communicative goals. One tendency in English and other languages is that important information, such as establishing the identity of a central discourse participant, is coded with syntactic forms which draw attention to it, such as the sentence subject (Bates and MacWhinney, 1989). How does this tendency to signal importance with sentence-subject coding interact with the subordinate-matrix asymmetry? Speakers place the context or supporting material (the subordinate clause) before the main assertion to produce an effect: it is placed first so that when the information in the main clause is encountered, it is evaluated appropriately. What happens if the supporting material needs to refer to *Bill?* This will be the case if the point of the supporting material is to situate *Bill* in space or time, or to describe some circumstantial or otherwise relevant event. If the topic is identified with a full lexical noun in the supporting material, then the situating force of this material is lessened. Instead of scene setting, the initial clause will be construed as having the

[1]Discourse topic is not to be equated with sentence subject. In naturally occurring text, the discourse topic is often so obvious and available that it is either not explicitly mentioned, or it is pronominalized (Givon, 1984). In interpreting sentences which do not have a prior context (such as the first sentence in a text such as a newspaper article) speakers must use what surface cues are available to determine the topic. Sentence subject is the surface device most commonly used to rapidly establish or re-establish a topic.

function of introducing a new discourse character. Example (10), as well as other sentences from Carden's (1982) corpus, illustrate this point.

(10) While *he* hadn't read the Gifford article, *Associate Dean of Yale College Martin Griffin* said that the 'best administrators are scholars', and that... (*Yale Daily News*, 31, Jan. 78).

Our hypothesis that a subordinate clause licenses backwards pronominalization by virtue of a backgrounding or scene-setting function is complemented by Bolinger's (1979) discussion of loose and tight connectives. He notes that connectives such as *and, although, before,* and *when* are "loose" connectives: they allow a change in agent or topic, and, when such adverbials are sentence-initial, they are typically accompanied by an intonation break. It is most natural for a speaker to reidentify the referent of a pronoun after a break of some kind.

Thus far, the system of explanation we have developed makes the same blocked vs. free predictions as Reinhart's c-command description: backwards reference is blocked if the pronoun is the sentence subject, but free if it is in an initial subordinate clause. In the next section, we turn to a case where c-command incorrectly predicts blocked coreference. We argue that backwards pronominalization is possible in sentences such as (8) because the imperfective or anterior aspect serves to background the initial clause with respect to a clause encoded with the simple past tense.

THE SIMPLE PAST VERSUS IMPERFECTIVE ASPECT

In this section, we investigate a case of backwards pronominal reference which has not received explicit linguistic analysis (although see Bolinger, 1979; Bosch, 1983). Although the pronoun is the sentence subject in examples (11)-(13), either an intrasentential or an extrasentential reference appears to be possible.

(11) He had been staring at the control panel for over an hour when Jack received a message from his commander.

(12) She had just reached the door to her apartment when Lois heard a shriek from the street.

(13) He was busy looking over the stolen exam when Michael heard someone opening the door.

In English, the simple past tense is typically used to denote discrete, punctate, completed events, as in (14). Events which are encoded with the simple past tense are typically construed as a narrative sequence in which the order of mention reflects the real world order of occurrence. An imperfect marker such as the progressive tense *ing* can be used to indicate that one event overlaps another, as in (15). The past perfect marker *had* signals that an event was completed prior to some reference point, as in (16). We will refer to these types of verbal aspect as **marked** to distinguish them from clauses containing the simple past tense.

(14) He walked to the store, decided what he wanted, and returned home.

(15) Walking to the store, he decided what he wanted. Upon returning home...

(16) He had walked to the store before, but this time...

When narrating a story, speakers need to convey to their listeners which events are part of the story line, and which events are to be interpreted as *comments* or *amplifications* of the story line. Drawing on an analysis of speakers of 20 different languages describing a 6 minute silent film, Hopper (1979) argues that the language of the story line (what he calls the **foreground**) differs from the language of the supporting material (the **background**). One important device for signaling the narrative foreground is the use of the completive aspect (the simple past tense in English). In contrast, when one event is simultaneous with another, or when an event is static or descriptive, the event can be construed as narrative background.

The sentences in (11)-(13) were problematic for c-command because the subject of the sentence is a pronoun, and thus c-commands all other nodes in the tree. But from the perspective of Hopper's analysis of verb aspect in discourse, the initial clause of each of these sentences is a backgrounded event. We

thus hypothesize that coreference is possible in these sentences for the same reasons that coreference is possible with sentence-initial subordinate clauses: because the initial clause functions as the setting or context for evaluating the material in the following clause.

SPEAKER JUDGEMENTS

In the previous section we demonstrated that the canonical communicative functions of clause type (matrix vs. subordinate), NP type (pronoun vs. full noun), and verbal aspect (simple past versus marked aspect) could be related to our linguistic intuitions about coreference assignment. Cases of marked aspect are unusual: backgrounding is achieved in a matrix clause by describing an action with imperfective or anterior aspect. Would a survey of speakers, questioned on a substantial number of such sentences, support our intuitions that verbal aspect functions to background the event, thus allowing backwards pronominalization? Is manipulating aspect as valid or strong a cue to backgrounding as syntactic subordination?

EXPERIMENTAL MATERIALS

In the exploration of constraints on intrasentential coreference, linguists typically make judgements of blocked versus free coreference for single, isolated sentences. As mentioned previously, the most common function of pronouns is to refer back to already established discourse entities (Givon, 1984). A very stringent test of speakers' willingness to allow a pronoun to precede its antecedent would thus be to give speakers a choice between discourse characters occurring before as well as after the pronoun.

We composed 61 three-sentence passages in which the first sentence introduced two characters and the second sentence topicalized one of them. These two sentences functioned as the discourse context for the third sentence, called the **target.** For each of the 61 passages, there were three versions of the target sentence, designed to corresponded to the sentence types in (7)-(9). In the **matrix** sentences, the initial clause was not introduced by an adverbial, and occurred in the simple past tense. In the **subordinate** cases, the initial clause was prefaced with an adverbial and contained simple past tense. In the **aspect**, the initial clause was the matrix clause of the sentence, and contained anterior or imperfective aspect.

CONTEXT	**THREE VERSIONS OF TARGET SENTENCE**
Kenneth and Andrew helped prepare each other for the big sumo wrestling match. Andrew weighed in at 193 kilos.	He put on the wrestler's outfit just as... (**matrix**) As he put on the wrestler's outfit... (**subordinate**) He had just put on the wrestler's outfit when... (**aspect**)
	... Kenneth/Andrew learned the match was canceled.

In each target sentence, either of the two discourse characters could be named. In the **relexicalization condition** the character who had been established as the discourse topic in the second sentence was renamed in the target sentence. In the **new name** condition, the non-topic was lexicalized in the target sentence.

Thirty-four raters were asked to read the 61 passages (along with filler passages) while sitting at a computer display. Each subject saw only one version of a passage. A question about the target sentence and a list of response options appeared simultaneously with each passage. For the passage above, the question and response options were as follows:

Question:	Who got into wrestling gear?						
Options:	Kenneth Best	Kenneth Better	Either	Andrew Better	Andrew Best	Someone Else	Ungrammatical

An additional 36 raters were asked to make reference judgements to just the target sentence from the 61 passages (the **no context** condition). Instead of being seated at a computer display, raters were

given a paper-and-pencil questionnaire. Because only the target sentences were to be rated, the response options included only one proper name. Response options for the target sentence described above were as follows:

Options:	Kenneth Best	Kenneth Better	Either	Someone Else	Ungrammatical

Our two factors were thus sentence type (three levels: **matrix, aspect, subordinate**) and discourse condition (three levels: **relexicalization, new name, no context.)**

HYPOTHESES

The **aspect** sentence type resembles the **matrix** in syntactic form, but resembles the **subordinate** in terms of its foreground-background structure. Will coreference judgements to the **aspect** targets be more similar to the pattern for the **subordinate** case or the **matrix** case? Our hypothesis is that the **aspect** sentences will be midway betweeen the two. The reason for this is that these sentence types contain conflicting cues: the sentence-initial pronoun is a cue to extrasentential reference, while the verbal aspect, as a cue to backgrounding, signals the listener to delay pronominal selectional until its relevance of the following clause is assessed.

We predict that the **no-context** condition will elicit a higher frequency of intrasentential referents than the **relexicalization** and **new name** conditions. This prediction comes from our characterization of the discourse function of pronouns: pronouns are used to refer to established or known discourse entities. In the absence of syntactic or semantic cues that the pronoun refers to a following NP, raters will try to find an extrasentential referent for the pronouns in our sentences. But without a previous context, raters will have to construct a context and populate it with a candidate referent. The "principle of least effort" suggests that raters will want to *minimize* the number of hypothesized discourse entities (Prince, 1981). Without a context, speakers will, if at all possible, attempt to find a reading of the sentence in which the intrasentential discourse character is the referent for the pronoun.

If this prediction is supported, it implies that judgements of blocked vs. free reference for sentences in isolation provide an inflated estimate of the conditions under which backwards pronominal reference is possible. Including a **no context** condition in the current study allows us to compare judgements of intrasentential reference for our three sentence types under two different discourse environments: an environment which encourages extrasentential reference (the **new name** condition), and an environment which encourages intrasentential reference (the **no context** condition).

RESULTS

For each passage, we calculated the percent of judgements in which raters decided that something *other* than the intrasentential proper name was the referent for the pronoun. Selection of the non-intrasentential proper name, or the "someone else" response, or the "ungrammatical" response was coded as a judgement that coreference between the pronoun and intrasentential NP was blocked (or non-preferred). ("Ungrammatical" responses were rare: only four passages received more than six such judgements.)

Figure 1 shows the percent of passages in which raters judged intrasentential reference to be blocked or non-preferred. An item-analysis anova revealed highly reliable main effects for sentence type, $F(2,120) = 72$, $p < .001$, and discourse condition $F(2,120) = 201$, $p < .001$, as well as a sentence type X discourse condition interaction, $F(4,420) = 5.2$, $p < .001$.

It is not surprising that syntactic differences in sentence type and discourse condition result in significantly different reference judgements. Of greater interest is quantifying how the *interaction* of cues affects reference selection. Within the **no context** condition, **matrix** sentence types received more blocked coreference judgements than the **aspect** sentences, $F(1,60) = 18$, $p < .001$, and **aspect** sentences

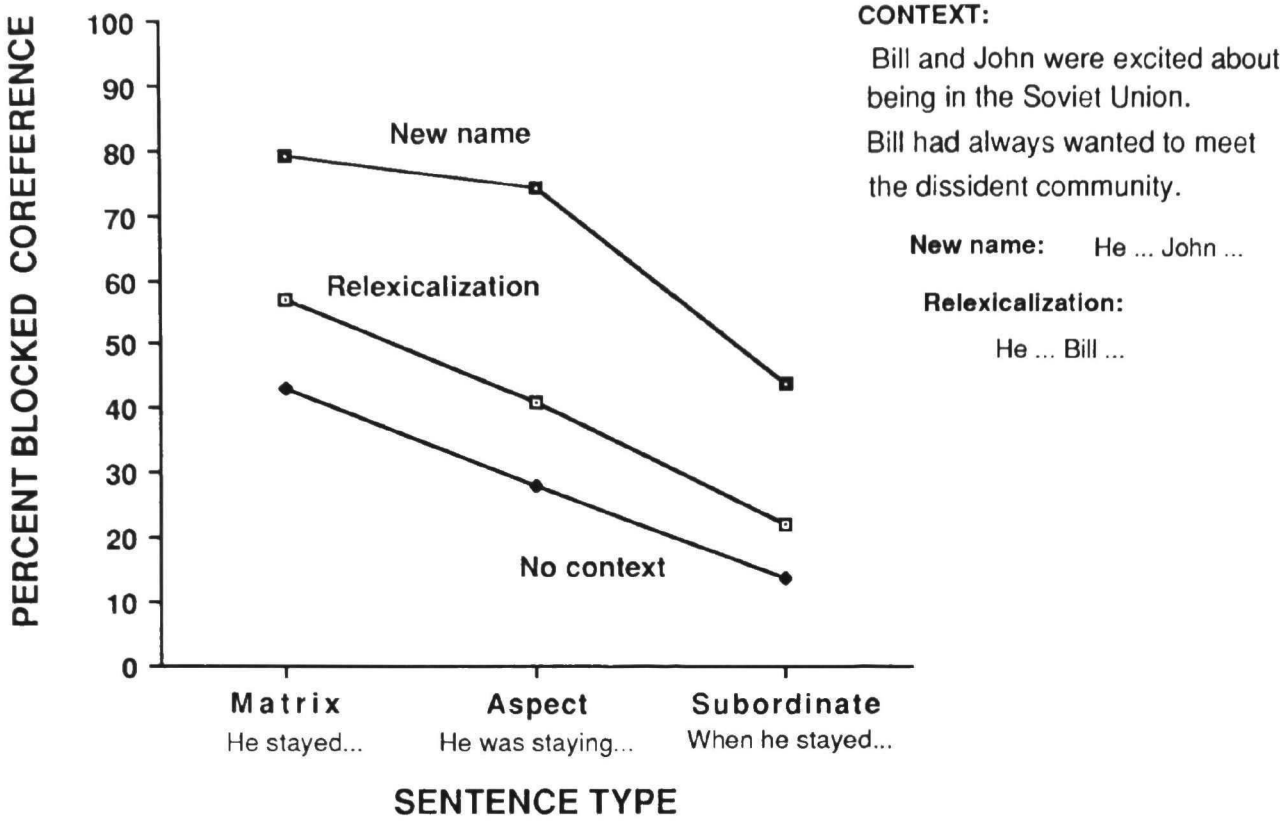


FIGURE 1: Percent of target sentences in which raters judged intrasentential reference to be blocked or non-preferred.

received more blocked judgements than **subordinate** sentences, $F(1,60) = 22$, $p < .001$. Comparable F values were obtained for the **relexicalization** condition. In the **new name** condition, however, the **aspect** sentences did not differ significantly from the **matrix** sentences, $F(1,60) = 2.1$, $p = .15$, although the **aspect** sentences did differ significantly from the **subordinate** sentences, $F(1,60) = 51$, $p < .001$.

DISCUSSION

In the first part of this paper we drew on work by functionalist grammarians to describe the communicative function of pronouns compared to lexical nouns, of subordinate clauses compared to matrix clauses, and of completive aspect compared to imperfective and anterior aspect. In the absence of other compelling syntactic or semantic information, listeners will look to the previous discourse context for a antecedent to a pronoun (Givon, 1984; Prince, 1981). What types of information would compel listeners to suspend this default strategy? We hypothesized that both syntactic subordination and durative/anterior aspect cause information to be perceived as background, context or supporting material. Because the listener knows that relevant information immediately follows a backgrounded clause, the default strategy of looking for an antecedent in the previous discourse may be relaxed.

Our survey of raters' judgements strongly supported our hypotheses. When provided with a backgrounding cue (either an aspect cue or syntactic subordination) subjects were more likely to allow a pronoun to precede its antecedent than when no backgrounding cue was present. As we predicted, however, aspect was not as strong a cue as syntactic subordination: in all discourse conditions, more intrasentential judgements were obtained for the **subordinate** sentences than for the **aspect** sentences.

What is the significance of the sentence type by discourse condition interaction? Recall that we analyzed the **aspect** sentences as containing two conflicting cues: the backgrounding cue signals that an

antecedent may be coming up, but coding the pronoun as the sentence subject signals that it refers to established material. In the **no context** and **relexicalization** condition, both cues appear to be contributing to reference judgements, since the percent of blocked/non-preferred intrasentential judgements for the **aspect** case is intermediary between the two other sentence types. Why, in the **new name** condition, was the **aspect** case indistinguishable from the **matrix** case? It appears that when raters are given a legitimate choice between an intrasentential and extrasentential referent, the aspect cue is not strong or salient enough to overwhelm the pronoun-in-subject position cue. However, in a conflict situation -- such as when speakers must decide whether relexicalization of a referent following pronominalization is possible, or must choose a referent in the absence of context -- then the aspect cue does succeed in allowing backwards pronominalization.

Functional approaches to understanding constraints on anaphoric reference have sometimes been represented as attempts to "indicate that coreference is not dependent at all on properties of the syntactic tree" (Reinhart, 1983, p. 94). Our own view -- supported by the data from speakers' judgements presented here -- is that the surface form of a sentence has a strong effect on referent selection. A more accurate characterization of the functionalist position is that speakers have communicative reasons for selecting a particular syntactic encoding (Givon, 1984; Kuno, 1987). An adequate characterization of the constraints on anaphoric reference will thus need to make reference to the communicative functions performed by syntactic forms. Furthermore, the interaction we found between discourse condition and sentence type suggests that syntactic cues do not operate in isolation from each other or from the discourse environment.

In the current paper, we showed that characterizing syntactic forms in terms of their communicative functions allowed us to predict a range of coreference judgements. Future work will be required to better understand the principles constraining the interaction of several syntactic cues and discourse contexts.

REFERENCES

Bates, E.A. & MacWhinney, B. (1989). Functionalism and the competition model. In B. MacWhinney and E. A. Bates (Eds.), *The crosslinguistic study of sentence processing*. New York: Cambridge University Press.

Bolinger, D. (1979) Pronouns in discourse. In T. Givon (Ed.), *Discourse and Syntax: Syntax and Semantics* **12**. New York: Academic Press.

Bosch, P. (1983). *Agreement and anaphora: A study of the role of pronouns in syntax*. New York: Academic Press.

Carden, G. (1980). Blocked forward anaphora: C-command the surface-interpretation hypothesis. Presented at the Fifty-fifth Annual Meeting of the Linguistic Society, of America, San Antonio.

Carden, G. (1982). Backwards anaphora in discourse context. *Journal of Linguistics*, **18**, 361-387.

Chafe, W. (1984). How people use adverbial clauses. *Berkeley Linguistic Society*, **10**, 437-449.

Chafe, W. (1988). Linking intonation rules. In J. Haiman and S. Thompson, (Eds.), *Clause combining in grammar and discourse*. Philadelphia: J. Benjamins.

Givon, T. (1984). *Syntax: A functional-typological introduction, Vol. 1*. Philadelphia: J. Benjamins.

Hopper, P.J. (1979). Aspect and foregrounding in discourse. In T. Givon, (Ed.), *Discourse and Syntax, Syntax and Semantics* **12**. New York: Academic Press.

Kuno, S. (1987). *Functional syntax: Anaphora, discourse and empathy*. Chicago: Chicago University Press.

Matthiessen, C., & Thompson, S. (1988). The structure of discourse and 'subordination.' In J. Haiman and S. Thompson, (Eds.), *Clause combining in grammar and discourse*. Philadelphia: J. Benjamins.

Prince, E. (1981). Toward a taxonomy of given-new information. In P. Cole, (Ed.), *Radical pragmatics*. New York: Academic Press.

Radford, A. (1988). *Transformational grammar*. Cambridge: Cambridge University Press.

Reinhart, T. (1983). *Anaphora and semantic interpretation*. London: Croom Helm.

A semantic analysis of action verbs based on physical primitives

Jugal K. Kalita and Norman I. Badler
Department of Computer & Information Science
University of Pennsylvania
Philadelphia, PA 19104-6389

Abstract

We develop a representation scheme for action verbs and their modifiers based on decompositional analysis emphasizing the implementability of the underlying semantic primitives. Our primitives pertain to mechanical characteristics of the tasks denoted by the verbs; they refer to geometric constraints, kinematic and dynamic characteristics, and certain aspectual characteristics such as repetitiveness of one or more sub-actions, and definedness of termination points.

1 Introduction

Suppose a human agent is asked to perform the following commands in a suitable environment:

- *Put the block on the table.*
- *Turn the switch to position 6.*
- *Roll the ball across the table.*
- *Open the door.*

Each of these sentences specifies an underlying task requested of an agent. In order to perform the task, the performing agent has to "understand" the command. Understanding the imperatives requires understanding the meanings of the action verbs such as *put, turn, open* and *roll*, and the meanings of prepositional words such as *on, in* and *across*. One has to integrate the meanings of the constituents and produce a meaning of the sentence as a whole taking pragmatic factors into consideration, wherever appropriate. Having done so, one has to construct a plan for execution of the task in the environment. Only then the agent may perform the action. All the above steps need to be followed even if the agent is not human, but program-controlled such as an animated agent in a computer graphics environment or a robotic agent.

Among the myriad issues involved in the comprehension of imperatives in a physical domain and the execution of the underlying tasks, this paper primarily deals with developing a representation for the meanings of verbs and prepositions in order to characterize underlying actions. Following Badler [Badler 1989b], we develop a representation in which movements denoted by action verbs can be decomposed into "primitives" with implementable semantics.

2 Identifying the nature of components

Case frames [Fillmore 1980], or its variations have been used extensively for representing the semantic roles between noun phrases and the verb in a clause. Comprehensive analysis of case frames have indicated limitations of their representational capabilities [Palmer 1985, Levin 1979, Jackendoff 1972]. These include the existence of a large number of exceptions, lack of consensus in establishing a universal set of cases, invalidity of multiple case assignments to a noun phrase, unsubstantiated semantic overloading of the names of cases, lack of explicit characterization of interrelationships among cases, and inadequacy of cases for representing the full complexity of concepts denoted by verbs. Considering the fact that a decompositional analysis obviates most of the above drawbacks [Palmer 1985], we opt for such an approach.

Palmer's research [Palmer 1985] in the domain of word problems in physics is similar to ours in objective and approach. However, our approach is "active", laying emphasis on implementability of the primitives, which is not the case in her study. Although her study considers some simple motion verbs, she treats them in a "static" fashion. Miller's analysis [Miller 1972] provided English paraphrases of some complex motion verbs in terms of "simpler" ones; Okada's analysis [Okada 1980] lacks in coherence and proper justifications. Schank's representation of verbs [Schank 1972], though impressive, suffers from the lack of uniformity in the levels of decomposition and explicit semantic anchoring of the primitives in terms of executability at a non-linguistic level.

Componential analysis performed here involves working from bottom up rather than from the language end. Our goal is to obtain representations as close to the physical world as possible. We obtain the components of the verbs

from an analysis of the "real" actions they represent taking into consideration physical attributes only. Thus, we use the linguistic technique of componential analysis for the representation of verbal meanings, but confine ourselves to consideration of mechanistic components only. This view of verbal semantics, albeit incomplete, is not short-sighted; such a study has not been carried out in an elaborate fashion to date and is vitally essential for understanding the physical nature of actions.

The attributes we identify for the characterization of the nature of tasks denoted by action verbs deal with the following criteria: kinematic/dynamic distinctions; the type and operational nature of relevant geometric constraints, aspectual considerations such as repetitiveness, whether termination condition(s) for the verb's underlying task are naturally well-defined, or need to be inferred contextually, etc. In prevalent characterizations of actions, kinematic and dynamic characteristics of verbs have not been considered in detail. Badler mentions these attributes, but does not elaborate [Badler 1989b].

2.1 Geometric constraints

Geometric constraints can be used to clearly describe certain movements and configurations of physical objects. They provide information regarding how one or more objects or sub-parts of objects relate to one another in terms of physical contact, absolute or relative location, inter-object distance, absolute and relative orientation, or path of motion. Nelson [Nelson 1985], Rossignac [Rossignac 1986] and Barzel [Barzel and Barr 1988] use geometric constraints to describe the motion of "simple" objects. Badler discusses specification of motions through constraints for succinct expression of activities of articulated human figures composed of hierarchic sub-parts performing natural actions or tasks [Badler 1989b, Badler 1989a].

2.1.1 Constraint types

Some verbs' underlying actions can be primarily described by positional or orientational constraints.

1. *Positional constraints*: This refers to situations in which a 0-, 1-, 2- or 3-dimensional object is constrained to a 0-, 1-, 2- or 3-dimensional region of space. For example, in order to execute the command *Put the ball on the table*, an arbitrary point on the surface of the ball has to be brought in contact with (or constrained to) an arbitrary point on the surface of the table. Another example is seen in the action underlying the imperative *Put the block in the box* where one needs to constrain the block (or the volume occupied by the block) to the interior volume of the box.
2. *Orientational constraints*: Consider the meaning of the preposition *across* in the sentence *Place the ruler across the table*. The interpretation of the preposition involves several components, one of which requires that the longitudinal axis of the ruler and the longitudinal axis of the table top be perpendicular to each other. This requirement can be expressed in terms of an orientational constraint.

2.1.2 Constraint operation

Verbs dealing with constraints can be classified considering whether they denote establishment, removal, maintenance or modification of (existing) geometric constraints.

1. Verbs whose central action requires that constraints established continue to hold: attach, hold, engage, fix, grab, grasp, hook.
2. Verbs whose central action requires that already existing constraints *cease* to hold. Examples include: detach, disconnect, disengage, release.
3. Verbs which refer to modification of an already existing constraint: loosen, tighten. We do not discuss such verbs in this paper.

2.2 Aspectual components

Aspect is an inherent semantic content of a lexical item pertaining to the temporal structure of the situation denoted by the lexical item, independent of context [Passonneau 1988]. Nakhimovsky's notion of aspectual class refers to internal temporal structuring of generic situations [Nakhimovsky 1988]. Moens's use of the term aspectual category is different in that it takes speaker's perspectives into consideration [Moens and Steedman 1988]. Below, we consider two aspectual characteristics: repetitiveness and telicity.

2.2.1 Repetitiveness or frequentation

1. Verbs whose underlying tasks definitely need repetitions of one or more sub-actions: roll, calibrate, screw, scrub, shake, rock.
2. Verbs for which repetitions may or may not be performed (i.e., whether repetitions are performed depends on the object(s) involved, information gathered from linguistic input, etc). Some example verbs are: cut, fill, lace, load, tamp.

2.2.2 Terminal conditions

We can divide the tasks identified by verbs into two categories based on this criterion. Nakhimovsky [Nakhimovsky 1988] also discusses the telicity property of actions denoted by verbs and verbal phrases.

1. **Atelic verbs**: The tasks denoted by such verbs do not have properly defined end conditions. The termination point may be determined by accompanying linguistic expressions or by context-dependent, task-dependent criteria such as default resulting states of affected object(s), or obtained through reasoning from commonsense knowledge, or knowledge of the goal to achieve, or defined by explicit feedback from simulation. For example, *cut* is atelic because an object can be cut and recut repeatedly, unless the linguistic imperative specifies that the object be cut once, or twice, etc., or specifies the size of the resultant pieces. Other examples of such verbs are: fold, hold, press, scrub, shake.
2. **Telic verbs**: These are verbs whose underlying tasks have properly defined built-in terminal points that are reached in the normal course of events and beyond which the processes cannot continue. Some examples are: align, assemble, attach, close, detach, drop, engage, fix, fill, place, release.

2.3 Kinematic—dynamic characterization of actions

Badler [Badler 1989b] states that an approach to movement (action) representation should be able to characterize (qualitative) specifications of kinematic and dynamic information, whenever appropriate. Dynamics describes the force or effort influencing motion. Kinematics deals with direct path or goals, and motion specification. Often movements along the same spatial path and toward the same spatial goal may be represented by different verbs such as *touch, press* and *punch*. These distinctions can be formulated in terms of dynamic specification.

1. *Kinematic:* These are verbs whose underlying actions can be expressed as a movement along an arbitrary path or at an arbitrary velocity. Some examples are turn, roll, rotate.
2. *Dynamic:* For a verb in this category, its underlying action can be characterized by describing the force which causes it. Examples include: push, shove, pull, drag, wring, hit, strike, punch, press.
3. Both kinematic and dynamic: These are verbs which have strong path as well as force components: swing, grip/grasp (vs. touch), twist.

3 Obtaining a representation for the verbs: primitives

The primitives we use are anchored outside the linguistic domain. Currently, we work in the domain of graphical animation and hence, our primitives are executable directly in a graphics environment called JACK [Phillips 1988]. The expression of linguistic semantics in terms of tested primitives vindicates the usefulness and "completeness" of our approach. Implementation of the dynamic primitives used can be found in [Otani 1989]. They pertain to forces, torques, mass and balance. We do not discuss dynamic primitives here since they are not directly relevant to the examples in this paper. Implementation of the primitives concerned with geometric relations are discussed in [Zhao and Badler 1989].

3.1 Geometric relations and geometric constraints

We specify geometric relations in terms of the following frame structure.

Geometric-relation: spatial-type:
source-constraint-space:
destination-constraint-space:
selectional-restrictions:

Spatial-type refers to the type of the geometric relation specified. It may have one of two values: ***positional*** and ***orientational.*** The two slots called ***source-constraint-space*** and ***destination-constraint-space*** refer to one or more objects, or parts or features thereof which need to be related. For example, in order to execute the command ***Put the cup on the table***, one brings the bottom surface of the cup into contact with the top surface of the table. The command ***Put the ball on the table*** requires bringing an arbitrary point on the surface of the ball in contact with the surface of the table top. Since, the items being related may be arbitrary geometric entities (i.e., points, surfaces, volumes, etc.), we call them ***spaces***; the first space is called the source space and the second the destination space. The slot ***selectional-restrictions*** refers to conditions (static, dynamic, global or object-specific) that need to be satisfied before the constraint can be executed.

Geometric constraints are geometric goals; they are specified as follows:

Geometric-constraint: execution-type:
geometric-relation:

Geometric constraints are of four types. They are distinguished by the ***execution-type*** component. The execution class or type of a constraint may be ***achieve, break, maintain*** or ***modify.***

3.2 Kinematics

The frame used for specifying the kinematic aspect of motion is the following:

Kinematics: motion-type:
source:
destination:
path-geometry:
velocity:
axis:

Motions are mainly of two types: *translational* and *rotational.* In order to describe a translational motion, we need to specify the source of the motion, its destination, the trajectory of the path, and the velocity of the motion. In the case of rotational motion, the path-geometry is always circular. The velocity, if specified is angular. An axis of rotation should be specified; otherwise, it is inferred by consulting geometric knowledge about the object concerned.

3.3 Kernel actions

The central part of an action consists of one or more components: ***dynamics, kinematics*** and ***geometric-constraints***—along with control structures stating aspectual or other complexities involved in the execution of an action. The constructs we use in the paper are: *repeat-arbitrary-times* and *concurrent.* The keyword ***concurrent*** is specified when two or more components, be they kinematic, dynamic or geometric constraints, need to be satisfied or achieved at the same time. The keyword *repeat-arbitrary-times* provides a means for specifying the frequentation propery of certain verbs. The verbs' semantic representation need not specify how many times the action or sub-action may need to be repeated. However, since every action is presumed to end, the number of repetitions of an action will have to be computed from simulation (based on tests for some suitable termination conditions), or by inference unless specified linguistically as in *Shake the block about fifty times.*

3.4 Representation of verbal and sentential meaning

Since our meaning representation is verb-based, the template for the representation of the meaning of a verb is also the frame for representation of meanings of sentences. The representation for a sentence has the following slots.

Verbal-representation: agent:
object:
kernel-actions:
selectional-restrictions:

Selectional restrictions may refer to dynamic or static properties of objects or the environment.

4 Some example verbs

4.1 A verb of establishment of positional constraint: *put*

Webster's dictionary [Woolfe 1981] defines one sense of the meaning of the verb *put* as *to place in a specified position or relationship*. We consider only the positional aspect of the meaning to obtain a lexical definition. The nature of the positional constraint is dependent upon the nature of the object, and the location specified by a locative expression. The lexical entry is

```
put (l-agent, l-object, l-locative) ⟵ agent: l-agent
                                      object: l-object
                                      kernel-action:
                                           geometric-constraint:
                                                execution-type:achieve
                                                spatial-type:   positional
                                                geometric-relation: l-locative
```

This representation tells us that *put* requires us to *achieve* a *positional* constraint between two objects, parts or features thereof. It does not indicate the type of positional relation to be achieved. The details of the geometric relation to be achieved have to be provided by the locative expression used which may be in terms of prepositions such as ***in***, ***on*** or *across*.

4.2 A kinematic verb: *roll*

The verb *roll* refers to two motions occurring concurrently: a rotational motion about the longitudinal axis of the object and a translational motion of the object along an arbitrary path. The rotational motion is repeated an arbitrary number of times. The representation for the verb *roll* we use is as follows:

```
roll (l-agent, l-object, path-relation)←
        agent: l-agent
        object:l-object
        kernel-action:
             concurrent { { kinematic:
                                  motion-type:   rotational
                                  axis:          longitudinal-axis-of (l-object)
                            } repeat-arbitrary-times }
                            { kinematic:
                                  motion-type:   translational
                                  path:          path-relation } }
        Selectional Restrictions:  has-circular-contour (l-object, longitudinal-axis-of (l-object))
```

4.3 A verb that removes constraints: *open*

We consider just one sense of *open*—the sense defined by Webster's Dictionary [Woolfe 1981] as *to move (as a door) from closed position*. The meaning is defined with respect to a specific position of the object under consideration. The closed position of the object can be viewed as a *constraint* on the position or orientation of the object. Thus, *open* can be considered as a verb whose underlying task undoes an existing *constraint*. The object under consideration is required to have at least two parts: a solid 2-dimensional part called the *cover* and an unfilled 2-dimensional part defined by some kind of frame: the *hole*. The meaning must capture that the agent performs an action whose result is to remove the constraint that object's cover and its hole are in one coincident plane. Additionally, the object's cover must occupy the total space available in object's hole in the constrained position. This is fulfilled by requiring that the two (sub-)objects (the hole and the cover) are of the same shape and size.

The definition for *open* is:

```
open (Ag, Obj) ← agent: Ag
                 object: Obj
                 kernel-action:
                      geometric-constraint:
                           execution-type:break
```

spatial-type: positional
geometric-relation: source-constraint-space: Obj • hole
destination-constraint-space: Obj • cover
Selectional Restrictions: contains-part (Obj, hole)
contains-part (Obj, cover)
area-of (Obj.cover) = area-of (Obj • hole)
shape-of (Obj.cover) = shape-of (Obj • hole)

5 Representing meanings of prepositions

In order to provide precise, implementable meanings of prepositions, we have been influenced by Badler [Badler 1975] and Gangel [Gangel 1984]. The semantics of the linguistic locative expression has also been discussed by Herskovits [Herskovits 1986]. Talmy's work [Talmy 1983] on the relation between language and space also discusses semantic representation of prepositions. We obtain lexical entries for a selection of prepositions in [Kalita 1990]. We include one such example below. Our definitions are limited in that they work with simple, solid, non-deformable geometric objects.

5.1 A representation for *on*

Among the senses of *on* defined by Herskovits [Herskovits 1986], the one we are interested in is: ***spatial entity supported by physical object.*** Examples of its use are seen in sentences such as *Put the block on the table* and *Put the block on the box.* The support can come in various forms as discussed by Herskovits. One situation which is commonplace or prototypical, is the one in which the located object rests on a free, horizontal, upward facing surface of the reference object; this need not be a top surface of the reference object, though it almost always is an *outer* surface (otherwise *in* is preferred). Webster's dictionary [Woolfe 1981] defines this meaning as *a function word to indicate a position over and in contact with.* We describe this meaning of *on* as

on (X,Y) ⟵ geometric-relation:
spatial-type: positional
source-constraint-space: any-of (self-supporting-spaces-of (X))
destination-constraint-space: any-of (supporter-surfaces-of (Y)))
selectional-restrictions:
horizontal (destination-constraint-space)
equal ((direction-of (normal-to destination-constraint-space) "global-up")
area-of (source-constraint-space) ≤ area-of (destination-constraint-space)
free-p (destination-constraint-space)

Given a geometric object, the geometrical function *self-supporting-spaces-of* obtains a list containing surfaces, lines or points on the object on which it can support itself. For example, a cube can support itself on any of its six faces, and a sphere on any point on its surface. The function *supporting-surfaces-of* finds out the surfaces on an object on which other objects can be supported. The functions *direction-of, normal-to, horizontal-p* and *area-of* are self-evident. The two directional constants *global-up* and *global-down* are defined with respect to a global reference frame.

6 Processing the sentence *Put the block on the table*

The sentence consists of the action verb *put.* The object is specified as the subject of the sentence and the location as a prepositional phrase. The meaning of the whole sentence, obtained by composing the meanings of its constituent parts [Kalita 1990] as specified by the definitions for *put* and *on*, is

agent: "you"
object: block-1
kernel-action: geometric-constraint:
execution-type: achieve
spatial-type: positional
geometric-relation:
spatial-type: positional
source-constraint-space: any-of (self-supporting-spaces-of (block-1))

destination-constraint-space: any-of (supporter-spaces-of (table-1)))
selectional-restrictions:
horizontal-p (destination-constraint-space)
equal (direction-of (normal-to destination-constraint-space) "global-up")
area-of (source-constraint-space) $\leq$ area-of (destination-constraint-space)
free-p (destination-constraint-space)

In order to execute the action dictated by this sentence, the program [Kalita 1990] looks at the knowledge stored about the block to find a part or feature of the block on which it can support itself. It can be supported on any one of its faces and no face is more salient than any other for supporting purposes. A cube (the shape of the block) has six faces and one is chosen randomly as the support area. The program searches the knowledge stored about the table for a part or feature which can be used to support other objects. It gathers that the table's function is to support "small" objects on its top which is also horizontal as required by a selectional restriction. Finally, the system concludes that one of the sides of the cube has to be brought in contact with the top of the table. Herskovits [Herskovits 1986] gives a very general discussion on the properties or attributes of objects reqired for such inferences. A planning module is needed to make the interface between the semantic representation level and the animation level complete. Details of object knowledge representation and planning can be found in [Kalita 1990]. Finally, the simulation program YAPS [Esakov and Badler 1990] performs appropriate computations and inferences where necessary, taking into consideration many aspects such as detailed geometric knowledge, default states, and knowledge of strengths and dimensions of the various parts of the body of the agent, and perform a graphical animation of the task's execution.

7 Conclusions

We have demonstrated that operational meanings of action verbs and their modifiers can be represented in terms of components pertaining to aspectuals, constraints, and kinematic/dynamic characterization. For additional examples of decomposition, the reader is referred to [Kalita 1990].

We have implemented a system incorporating the semantic processing discussed in this paper in Common Lisp on a HP workstation. The semantic output is further processed in consultation with the detailed knowledge stored about the objects under consideration. Finally, this interpreted output is used by YAPS [Esakov et al 1989, Esakov and Badler 1990] to drive a graphical animation. The successful animation of tasks starting from natural language input provides a sound anchoring for our semantic representation.

In this paper, we have not discussed the classification of lingusitic arguments of a verb as obligatory or optional, and the treatment of different classes of arguments. This issue is discussed in [Kalita 1990]. We have not also considered significant aspects of language usage such as intention, non-geometric goals and beliefs of agents. The reason is that ours is an in-depth study of the purely physical aspects of actions; thus, direct physical realizability is of primary importance. Our study has to be complemented by augmenting it with such intensional considerations for a fuller understanding.

Acknowledgements

This research is partially supported by Lockheed Engineering and Management Services (NASA Johnson Space Center), NASA Ames Grant NAG-2-426, FMC Corporation, Martin-Marietta Denver Aerospace, NSF CER Grant MCS-82-19196, and ARO Grant DAAL03-89-C-0031 including participation by the U.S. Army Human Engineering Laboratory.

References

[Badler 1975] Badler, Norman I., 1975. *Temporal Scene Analysis: Conceptual Description of Object Movements.* Technical Report TR-80, University of Toronto, Toronto, Ontario. (University of Pennsylvania, Department of Computer and Information Science, Technical Report 76-4).

[Badler 1989a] Badler, N.I., 1989. Artificial Intelligence, Natural Language and Simulation for Human Animation. In Magnenat-Thalmann, M. and Thalmann, D. (editors), *State-of-the-Art in Computer Animation.* Springer Verlag, New York.

[Badler 1989b] Badler, N.I., 1989. A Representation for Natural Human Movement. In Gray, J.A. (editor), *Dance Technology: Current Applications and Future Trends.* AAHPERD Publications, Reston, VA.

[Barzel and Barr 1988] Barzel, R. and Barr, A.H., 1988. A Modeling System Based on Dynamic Constraints. *Computer Graphics* 22(4):179–188.

[Esakov and Badler 1990] Esakov, Jeffrey and Badler, Norman I., 1990. An Architecture for High-Level Human Task Animation Control. In Fishwick, P.A. and Modjeski, R.S. (editors), *Knowledge-Based Simulation: Methodology and Aplications.* Springer Verlag, New York.

[Esakov et al 1989] Esakov, J.; Jung, M.; and Badler, N. I., 1989. An investigation of language input and performance timing for task animation. In *Proceedings of Graphics Interface 89*, pages 86–93. Morgan-Kaufmann, Palo Alto, Ca.

[Fillmore 1980] Fillmore, C., 1980. The Case for Case. In Bach and Harms (editors), *Universals in Linguistic Theory*, pages 1–88. Holt, Reinhart and Winston, New York.

[Gangel 1984] Gangel, Jeffrey, 1984. A Motion Verb Interface to a Task Animation System. Master's thesis, Department of Computer and Information Science, University of Pennsylvania.

[Herskovits 1986] Herskovits, Annette, 1986. *Language and Spatial Cognition. Studies in Natural Language Processing.* Cambridge University Press, Cambridge, England.

[Jackendoff 1972] Jackendoff, R.S., 1972. *Semantic Interpretation in Generative Grammar.* MIT Press, Cambridge, Massachussetts.

[Kalita 1990] Kalita, J.K., 1990. Analysis of Some Actions Verbs and Synthesis of Underlying Tasks in an Animation Environment. Forthcoming Ph.D. Thesis, Department of Computer and Information Science, University of Pennsylvania.

[Levin 1979] Levin, B., 1979. Predicate-Argument Structures in English. Master's thesis, MIT.

[Miller 1972] Miller, G.A., 1972. English Verbs of Motion: A Case Study in Semantics and Lexical Memory. In Melton, A. W. and Martin, E. (editors), *Coding Processes in Human Memory*, pages 335–372. V.H. Winston and Sons.

[Moens and Steedman 1988] Moens, Marc and Steedman, Mark, June 1988. Temporal Ontology and Temporal Reference. *Computational Linguistics* 14(2):15–28.

[Nakhimovsky 1988] Nakhimovsky, A., June 1988. Aspect, Aspectual Class and the Temporal Structure of Narrative. *Computational Linguistics* 14(2):29–43.

[Nelson 1985] Nelson, G., 1985. Juno, a Constraint-based Graphics System. *Computer Graphics* 19(3):235–243.

[Okada 1980] Okada, N., 1980. A Conceptual Taxonomy of Japanese Verbs for Understanding Natural Language and Picture Patterns. In *Proceedings of ICCL*, pages 127–135.

[Otani 1989] Otani, Ernest, 1989. Software Tools for Dynamic and Kinematic Modeling of Human Motion. Master's thesis, Department of Mechanical Engineering, University of Pennsylvania, Technical Report, MS-CIS-89-43, Philadephia, PA.

[Palmer 1985] Palmer, Martha, 1985. *Driving Semantics in a Limited Domain.* PhD thesis, University of Edinburgh, Scotland.

[Passonneau 1988] Passonneau, R.J., June 1988. A Computational Model of the Semantics of Tense and Aspect. *Computational Linguistics* 14(2):44–60.

[Phillips 1988] Phillips, C.B., 1988. *JACK User's Guide.* Computer Graphics Research Laboratory, Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA.

[Rossignac 1986] Rossignac, J.R., 1986. Constraints in Constructive Solid Geometry. In *Proceedings of Workshop on Interactive 3-D Graphics*, pages 93–110. ACM.

[Schank 1972] Schank, Roger, 1972. Conceptual Dependency: A Theory of Natural Language Understanding. *Cognitive Psychology* 3(4).

[Talmy 1983] Talmy, Leonard, 1983. How Language Structures Space. In Pick, H. and Acredols, L. (editors), *Spatial Orientation: Theory, Research and Application*, pages 225–282. Plenum Press, New York.

[Woolfe 1981] Woolfe, Henry (editor), 1981. *Webster's New Collegiate Dictionary.* G. & C. Merriam Company, Springfield, MA.

[Zhao and Badler 1989] Zhao, J. and Badler, N.I., 1989. *Ral Time Inverse Kinematics with Joint Limits and Spatial Constraints.* Technical Report MS-CIS-89-09, Department of Computer and Information Science, University of Pennsylvania.

Some Principles of the Organization of Verbs in the Mental Lexicon

Christiane Fellbaum
Cognitive Science Laboratory
Princeton University

Roger Chaffin
Trenton State College

We tested the organization of verbs in semantic memory in terms of five semantic relations. These relations are modeled on, but different from, those commonly assumed to organize the noun lexicon. In a restricted association task, subjects were given 30 seconds to generate verb-only responses to a verb stimulus, and the responses were classified in terms of the five relations. 28 different verb stimuli were selected from pairs that had been identified as examples of the relations under study. When idiosyncratic responses were discounted, the five relations accounted for 94% of all responses. The dominant relation, accounting for about 25% of the answers, turned out to be troponymy, the hyponymic *manner-of* relation, which links verbs like *munch, eat,* and *consume*. The second most frequent responses (14.4%) were examples of entailment, as between *dream* and *sleep*, followed by synonymy (*shout-holler-yell*) with 13.5% of the answers, and opposition relations (such as *enter-exit*) with 8%. The least frequently generated responses (4.1% of the total) represented the presupposition relation (exemplified by *cure-treat.*) For verbs that have a "tree" structure with three or more lexicalized taxonomic levels, associations seem to be strongest between the superordinate and what might be a "basic" level, while higher-level verbs are rarely generated. The semantically more elaborate troponyms also enter into opposition relations with each other, unlike the verbs on the superordinate level. Some verbs have a relatively "flat" structure, and are linked only to antonyms and synonyms; the organization of these verbs, which tend to cluster in the *change* verb lexicon, resembles that found for adjectives. The results lend support to a model of the structure of the verb lexicon based on these relations.

1. Conceptual Relations Among Verbs as the Organizing Principle of the Mental Lexicon

The typical high school graduate knows upwards of 40,000 words (Miller, 1988.) The organization of such a large number of words in the mental lexicon is commonly assumed to be in terms of semantic relations connecting the words to each other (see Evens, 1988, for a summary.) Miller (1969) noted that much of the available association data (Kent and Rosanoff, 1910; Russell and Jenkins, 1954; and Woodrow and Lowell,

The authors gratefully acknowledge the assistance of the members of the Experimental Methods Class at Trenton State College, Spring, 1989.

1916) could be interpreted in the light of two semantic relations, hyponymy and meronymy. Hyponymy, a relation based on category membership, links words such as *robin* and *bird*, where the former can be said to be *a kind of* the latter. Words related by hyponymy are believed to be stored together (Collins and Quillian, 1969, Neisser, 1988, and others), making this relation a major organizer of the mental lexicon. Evidence also exists for an organization in terms of meronymic (or part-whole) relation, which relates words like *wheel-car* and *tree-forest* (Chaffin, Herrmann, and Winston, 1988; Winston, Chaffin, and Herrmann, 1987). Hyponymy and meronymy are relations that appear to be best fitted to nouns and noun concepts (Beckwith, Fellbaum, Gross, and Miller, to appear.) Less attention has been focused on the relations among other parts of speech. Gross, Fischer, and Miller (1989) showed that adjectives are organized in terms of antonymy and similarity relations, and that these relations hold between individual lexical items, rather than between entire concepts that can be expressed by more than one word.

The purpose of the present inquiry is to identify the major semantic relations between verbs by means of an association task.

Few attempts have been made to study the organization of verbs (but see Rifkin, 1985, and Rips and Conrad, 1989.) Fellbaum and Miller (to appear) suggest, contrary to Rips and Conrad (1989), that the organization of verbs differs substantially from that of nouns. Instead of hyponymy, they posit a manner, or "troponymy", relation (from Greek *tropos*, fashion or manner). Thus, *nibble*, *munch*, and *gorge* are troponyms of *eat*, in that they refer to manners of eating ("manner" here denotes a variety of semantic elements, such as speed, direction, location, time, intent, quantity, etc.)

The relation of troponymy has been extensively employed in the construction of WordNet, an on-line lexical database constructed on the basis of theories of human lexical organization (Miller et al., 1988; Beckwith et al., to appear; Fellbaum, ms.) Postulating this relation made it possible to cast the English verb lexicon into a tight network, but its usefulness in constructing such a network did not in itself constitute any evidence for the existence of troponymy as an organizer of the mental lexicon. The present study was intended to provide just such evidence.

The relation of synonymy has also been assumed to be a strong organizer of both the noun and the verb lexicon. Furthermore, we test the status of antonymous relations as organizers of the verb lexicon. Such pairs as *rise-fall*, *shout-whisper*, and *enter-exit* represent different kinds of opposition (Cruse, 1986; Lyons, 1977.) The verbs in each pair are always co-troponyms, i.e., daughters of the same superordinate, but they elaborate the concept expressed by that superordinate in contrasting ways. In such opposite pairs as *tie-untie*, one member refers to the undoing or reversing of the action denoted by the other member. Opposite pairs like *give-take* and *question-answer* are converse and symmetric, in that the action denoted by one member results in the action referred to by the other member, but performed by a different participant.

Three additional relations have been postulated as organizers of the verb lexicon. They are variations of a relation termed "entailment" by Fellbaum and Miller (to appear). It was argued there that this relation is the analog of the part-whole relation among nouns. In most cases, verbs denoting activities cannot be broken down into other verbs referring to sequentially ordered subactivities. The closest approximation to meronymy

among verbs can be found in such verb phrases as *write a paper*, denoting events or "script"-like activities (Schank and Abelson, 1977) that can be broken down into sub-activities like *submit*, *proofread*, etc. Note that these component activities tend not to be lexicalized, but are referred to by entire verb phrases. To derive "parts" of most English verbs, one could undertake a semantic decomposition. However, this approach would, in most cases, not yield verbs but, rather, such components as causation (cf. the celebrated example of *kill*,) negation, and aspect. While these meaning components often have a morphological surface realization (such as affixes of various kinds,) they can usually not be expressed as independently lexicalized verbs. The "entailment" relation that Fellbaum and Miller postulate instead is based on the notion of entailment, or strict implication, in logic, where a proposition P is said to entail a proposition Q iff there is no conceivable state of affairs that could make P true and Q false. Entailment here denotes the relation between two verbs V_1 and V_2 that holds when the statement *Somebody* V_1*-s* entails the statement *Somebody* V_2*-s*. For example, entailment relates such verbs as *snore* or *dream* and *sleep*, and also *drive* and *ride*, where the former activity always entails, and overlaps temporally with, the latter (i.e., you cannot snore or dream without sleeping at the same time.)

Another, similar, relation is a kind of backward presupposition; this relation is illustrated by such pairs as *succeed-try* and *digest-ingest*. Unlike the verbs in the entailment relation, the verbs in this relation are not linked by temporal inclusion: you must have performed the presupposed action prior to the presupposing one.[1]

The purpose of the present study was to see whether the semantic relations described above (synonymy, troponymy, antonymy, entailment, and presupposition) really do serve to organize people's verb lexicon. Subjects' generation of verbs associated with a stimulus verb shows whether the relations of the stimulus and the subject-generated verbs are of the kind we have postulated here. Another study, which we will report on separately, tests subjects' ability to recognize our hypothesized relation by distinguishing one verb pair from a set of pairs as being an instance of that relation and as differing from other relations. This has been shown to be the case for hyponymy and meronymy among nouns. People recognize that *robin:bird* and *oak:tree* are examples of the same relation, and that these pairs differ from those like *neck:giraffe* and *petal:flower* (Chaffin and Herrmann, 1988a.) In the present study, we tested subjects' ability to produce verbs related in the five ways outlined above.

2.0 The Restricted Association Task

Our specific aim was to test the reality of the semantic relations between verbs that underlie the structure of WordNet. In the association task that we report here subjects were asked to restrict the associations they produced to verbs. This restriction is

[1] Verbs linked by troponymy, such as the pair *traipse-walk*, are always linked by a (temporally including) entailment relation: a verb referring to the elaboration of another verb always entails the unelaborated verb. Similarly, entailment always accompanies certain kinds of opposition or antonymy (e.g., both members of the pair *enter* and *exit* entail *walk*.) And some verb pairs are linked by both an opposition and a presupposition relation: *tie-untie*. These secondary relations are not the subject of our study here.

somewhat artificial in that people clearly do not form mental associations between words and concepts represented by one type of syntactic category only. We also performed an unrestricted association task experiment, where subjects' responses to verb stimuli were not restricted to verbs; the results of this study, which will be reported on elsewhere, should reflect the structure of the mental lexicon more accurately.

2.1 Method

Eleven subjects were each given 28 verbs as stimuli. The 28 verbs were chosen from pairs representing typical examples of the five different relations used to code verbs in WordNet (synonymy, antonymy, troponymy, entailment, and presupposition.) From each such pair, only one verb was chosen. All of the stimuli are words occurring relatively frequently in the language (X=32.4, Francis and Kucera, 1982.) Thus, *rise* was chosen from the pair *rise-fall*, coded in WordNet as opposites, and *waltz* was selected from the pair *waltz-dance*, illustrating the relation of hyponymy (troponymy). Besides its one prominent relation to another verb, each verb is usually connected further to other verbs, and we expected the different responses to show these diverse relations.

Eleven students in an Experimental Methods Class at Trenton State College were each given a note pad on which they were instructed to write down all the verbs that came to their minds after the stimulus verb had been read aloud. They were told to use a different page to record their responses to each stimulus. For each stimulus, they were given 30 seconds to respond.

2.3 Results

The two authors independently rated the responses in terms of the five semantic relation holding between the stimulus and the response. For each relation, we had formulated an illustrative sentence with one slot each for the stimulus and the response verb. When appropriate, we differentiated between the cases where the response occurred either in the first or in the second slot in the sentence. For example, the sample sentence for the troponymy relation was *To____is to ___in some manner*. If the troponym constituted the response (in the first slot), the answer was rated as *T1*. If the superordinate term (in the second slot) was generated, the response was classified as *T2*. This directional distinction was also relevant in the cases of entailment and presupposition, but not for the "symmetric" relations of synonymy and antonymy. The overall agreement rate was 83.5%. Disagreements, which were resolved by discussion, were generally due to a coder's failing to recognize a low frequency sense of the response.

The average number of responses by the eleven subjects to each of the 28 stimuli was 36.85. Of these responses, an average of 23.96 per stimulus fell into one of the categories we had identified. Table 1 lists the frequencies with which the eleven subjects responded to each stimulus word. The frequencies are means taken across the 28 different stimulus words. Frequencies are given separately for each of the five relations under study, totalled across the five relations, and for responses that could not be classified in terms of the five relations. The relations under study accounted for 65% of all responses. The first row of the table gives the frequencies for all responses. When idiosyncratic responses were eliminated by looking only at words generated by more than

two subjects, the proportion of responses accounted for by the relations under study rose to 94%. Frequencies for words given by more than two subjects are listed in the second row of the table.

Inspection of the top row of Table 1 shows that the relation that appeared most frequently in the subjects' responses was troponymy (25.1%), with entailment (14.4%), synonymy (13.5%), and antonymy or opposition (8%) appearing with intermediate frequencies, and presupposition (4.1%) having the lowest frequency. These differences were significant, $F(4,108)=6.43$, $p<.001$. The ordering of frequencies was the same for words given by more than two subjects, shown in the second row of Table 1, $F(4,108)=4.86$, $p<.001$. The ordering of relations was also largely the same when only the first response to each stimulus was counted. The frequencies for first responses are shown in the third row of Table 1. Again, troponymy accounted for the most frequent (27.0%), and presupposition for the least frequent (3.2%) responses. The ordering of the relations with intermediate frequencies differed from that for all responses. Synonymy appeared almost as frequently as troponymy (22.1%), with antonymy (15.9%) and entailment (13.3%) appearing somewhat less frequently. The difference in the frequency of the five relations for the first responses was reliable, $F(4.108)=3.38$, $p<.001$.

3.0 Discussion

The results of this experiment lends support to our hypothesis about the mental organization of the verb lexicon, in that 65% of the responses could be classified in terms of the semantic relations postulated; this figure rises to 94% when idiosyncratic answers given by less than two subjects are eliminated. These answers often denote verbs that are in a co-ordination relation with the stimulus (such as *read-write*), or co-troponyms of the stimulus (such as the responses *rumba, cha-cha,* and *mambo* given to the stimulus *waltz.*) The results permit a more fine-grained analyis of the structure of the verb lexicon and the distribution of some of the semantic relations within the verb lexicon.

3.1 Troponymy

Troponyms elaborate the concepts expressed by their superordinate by adding some fairly specific manner component, which, in a semantic decomposition, could be expressed by means of an adverb or an adverbial phrase. Lexicalization is richest on the subordinate level of the troponyms, because a number of manner elaborations are usually possible for a given superordinate. Opposition relations among verbs tend to be found only among the troponyms, where the oppositions derive from the manner elaboration (e.g., *gobble* and *nibble* constitute a pair of opposing co-troponyms of *eat.*) The same kind of manner relation does not exist between the superordinates (such as *eat*, *drink*, and *write*) and their respective superordinates (*consume* and *communicate*, respectively), which seem somewhat more "remote." The same turned out to be true for contact verbs, such as *hit* and *break*, which are rich in troponyms but tend not to have antonyms on the superordinate level. Fellbaum (ms.) notes the general infelicitousness of transivity statements involving verbs, which can be attributed to subtle differences in the relations between verbs that are separated by more than one level. On the analogy of Rosch's et al. (1976) important work on noun concepts, one might argue that the troponyms constitute

"basic-level" verbs. The manner elaborations that are part of the troponyms' elaborate semantics and whose differentiating function shows up in the opposition relations between the troponyms correspond to the large number of attributes characteristic of the basic level noun concepts studied by Rosch et al. Our results indicate that associations are strongest between the level of the troponyms and their superordinate (e.g., *munch* and *eat*,) rather than between the superordinate and its higher term (such as *eat* and *consume*.)

Subjects generated troponyms and superordinates with about equal frequency (12.6% and 12.3%, respectively.) The number of troponymic and superordinate responses to a particular stimulus depended on the level of the stimulus in its particular hierarchy. Some verbs do not have lexicalized superordinates, such as the verb *hit*, which elicited mostly troponyms like *bang, knock, punch*, and *slap*. Others do not have troponyms, such as *waltz*, which elicited its superordinate, *dance*, 11 times. In the cases of verbs that have three or more lexicalized taxonomic tiers, responses were generally limited to verbs on what might be termed the "basic" and "superordinate" levels, and subjects did not generate words from further "up" or "down" in the hierarchy. "Basic-level" verbs elicited most frequently their superordinates (for example, *sip* elicited *drink* 10 times,) while a superordinate, such as *eat*, elicited its respective superordinate "genus" term, *consume*, only once, and generated more troponyms, such as *gobble, gorge, dine, crunch, devour, binge*, and *stuff*.

3.2 Synonymy and Antonymy

While some verbs have the "vertical" structure with at least three taxonomic levels that we saw in the cases of *eat* and *drink*, there are other verbs whose structure seems to be flat, or "horizontal." These verbs are related to synonyms and verbs expressing an opposition, but they have no superordinate and few troponyms. The stimulus *shout*, for example, produced mostly synonyms, such as *bellow, yell, holler*, and only a few superordinates, such as *tell, voice, talk, interject* and *speak* (24 and 5 responses, respectively.) Another example is *close*, which generated both its antonym *open* and its synonym *shut* with equal frequency (10 responses each,) but only 5 hyponyms. Similarly, *respond* elicited most frequently a synonym *answer* (10 times) and an opposite term *question* (5 times); the superordinate term *talk* was given only twice. *Exhale* elicited its lexical and semantic opposite *inhale* (7 times,) and its entailed verb *breathe* (10 times.) *Enter* elicited its "clang" opposite, *exit*, most often, but also two other opposites, *leave* and *go*. *Enter* and *respond* exhibit a structure resembling the one found for adjectives (Gross et al., 1989), where two adjectives form a strong opposition, and where each of these "direct antonyms" in turn is related to its synonyms, which constitute "indirect antonyms." In WordNet, verbs with such a relatively "flat" structure tend to cluster in the *change* verb group, where opposition is heavily represented as a major relation in this part of the lexicon.

3.3 Entailment and Presupposition

The entailment relation with temporal overlap between the activities denoted by the two verbs was represented in a number of responses, such as *snore* and *dream* to the stimulus

sleep. Here, the stimulus constitutes the entailed verb, and the responses the entailing activity. By contrast, *steer* elicited the entailed verbs *drive* (four times); and *chase* most frequently generated its entailed activities *run* (7 times) and *follow* (6 times). In terms of a taxonomic structure, the subjects generally moved "up", rather than "down."

Finally, we also found evidence that the presupposition relation functions as an associative link. Subjects responded to the stimulus *marry* with the presupposed verbs *love* (4 responses) and *engage* (2 responses). *Win* elicited *try, compete*, and *gamble*. Other clear examples were the frequent responses *treat* and *medicate* for the stimulus *cure*. As in the examples of entailment, subjects generally moved "up" in the hierarchy, from presupposing to presupposed verb.

4.0 Conclusion

The results of our study give evidence for the mental organization of the verb lexicon in terms of semantic relations between verbs. The five relations that were specifically tested all appear to serve as links between verbs, with some relations playing a more prominent role within the verb lexicon than others. The responses elicited by our stimuli indicate that the *manner-of* relation, or troponymy, is the most important organizer of the verb lexicon, followed by entailment, synonymy, opposition relations, and presupposition.

References

Beckwith, R. C. Fellbaum, D. Gross, & G. A. Miller (to appear). WordNet: A Lexical Database Organized on Psycholinguistic Principles. To appear in: *Proceedings of the First International Workshop on Lexical Acquisition*, ed. U. Zernik. Hillsdale, NJ: Erlbaum.

Chaffin, R. and D. J. Herrmann (1988a). The nature of semantic relations: A comparison of two approaches. In M. Evens (1988), 289-334.

Chaffin, R. and D. J. Herrmann (1988b). Effects of relation similarity on part-whole decisions. *Journal of General Psychology*,**115**, 131-139.

Chaffin, R., D. J. Herrmann, and M. Winston (1988). An empirical taxonomy of part-whole relations: Effects of part-whole relation type on relation identification. *Language and Cognitive Processes*, **3**, 17-48.

Collins, A. M., & M. R. Quillian (1969). Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, **8**, 240-247.

Cruse, D.A. (1986). *Lexical Semantics*. New York: Cambridge University Press.

Evens, M. W. (ed.)(1988). *Relational Models of the Lexicon*. Cambridge: Cambridge Unversity Press.

Fellbaum, C. (ms). The English verb lexicon as a semantic net. Princeton University, Cognitive Science Laboratory.

Francis, W. N., and H. Kucera (1982). *Frequency Analysis of English Usage: Lexicon and Grammar*. Boston, MA: Houghton Mifflin.

Gross, D., U. Fischer, & G. A. Miller (1989). The organization of adjectival meanings. *Journal of Language and Memory*, **28**, 92-106.

Kent, G. H., and A. J. Rosanoff (1910). A study of association in insanity. *American Journal of Insanity*, **67**, 317-390.

Lyons, J. (1977). *Semantics*. 2 vols. New York: Cambridge University Press.
Miller, G. A. (1969). The organization of lexical memory: Are word associations sufficient? In: G. A. Talland and N. C. Waugh (eds.), *The Pathology of Memory*. New York: Academic Press.
Miller, G. A., C. Fellbaum, J.Kegl, & K. Miller. (1988). WordNet: An electronic lexical reference system based on theories of lexical memory. *Revue quebecoise de linguistique*, **17**, 181-213.
Miller, G. A. (1988). The challenge of universal literacy. *Science*, **24**, 1293-1299.
Neisser, U. (ed.) (1988). *Concepts and conceptual development: Ecological and intellectual factors in categorization.*. Cambridge: Cambridge University press.
Rifkin, A. (1985). Evidence for a basic level in event taxonomies. *Memory and Cognition*, **13**, 538-556.
Rips, L. J., & F. G. Conrad (1989). Folk psychology of mental activities. *Psychological Review*, **96**, 187-207.
Rosch, E., C. B. Mervis, W. Gray, & P. Boyes-Braem (1976). Basic objects in natural categories. *Cognitive Psychology*, **8**, 382-439.
Russell,W. A., and J. J. Jenkins (1954). *The complete Minnesota Norms for Responses to 100 Words From the Kent-Rosanoff Word Association Tests*. Mimeograph, Department of Psychology, Minneapolis: University of Minnesota.
Schank, R. C. & R. P. Abelson (1977). *Scripts, Plans, Goals, and Understanding*. Hillsdale, NJ: Erlbaum.
Winston, M. Chaffin. R., & D. J. Hermann (1987). A taxonomy of part-whole relations. *Cognitive Science*, **11**, 417-444.
Woodrow, H., and F. Lowell (1916). Children's association frequency tables. *Psychological Monographs*, **22**, No. 97.

Appendix

Table 1
Mean Response Frequencies Across Stimulus Words (N=28),
Classified by Relation, for Eleven Subjects

Relation of Response to Stimulus

	Synonym	Antonym	Troponym	Entail-ment	Presupp.	Total for all Relations	Other Responses
All Responses	4.93	2.93	9.18	5.46	1.50	23.96	12.89
Responses given by more than two subjects	3.61	2.39	5.89	3.86	.86	16.60	1.29
First Response	2.43	1.75	2.93	1.46	.36	8.93	2.07

Semantic Classification of Verbs from their Syntactic Contexts: Automated Lexicography with Implications for Child Language Acquisition

Michael R. Brent
MIT AI Lab
545 Technology Square
Cambridge, Massachussets 02139
michael@ai.mit.edu

Abstract

Young children and natural language processing programs share an insufficient knowledge of word meanings. Children catch up by learning, using innate predisposition and observation of language use. However, no one has demonstrated artificial devices that robustly learn lexical semantic classifications from example sentences. This paper describes the ongoing development of such a device. An early version discovers verbs with a non-stative sense by searching in unrestricted text for verbs in syntactic constructions forbidden to statives. Our program parses unrestricted text to the extent necessary for classification. Once the parsing is done recognizing the telltale constructions is so easy even a two-year-old child could do it. In fact, Landau and Gleitman (1985) and especially Gleitman (1989) argue that children must, can, and do use the syntactic constructions in which verbs appear to support meaning acquisition. In this paper we use our program to examine the difficulty of exploiting two particular syntactic constructions to discover the availability of non-stative senses, concluding that only very little sophistication is needed. This conclusion bolsters the general position of Gleitman (1989) that children can exploit syntactic context to aid in semantic classification of verbs.

1 Introduction

Young children and natural language processing programs face a common problem: everyone else knows a lot more about words. Children indisputably catch up by learning, using innate predisposition and observation of language use. However, no one has succeeded in creating artificial devices that robustly learn lexical classifications from example sentences. This paper describes the ongoing development of such a device. An early version discovers verbs with a non-stative sense by searching in edited text[1] for verbs in syntactic constructions forbidden to statives. To do this, it must partially parse the sentence, which in turn requires knowing the major syntactic categories of most of the words. Once the partial parsing is done, recognizing the telltale constructions is so easy even a two-year-old child could do it. In fact, Landau and Gleitman (1985) and especially Gleitman (1989) argue that children must, can, and do use the syntactic constructions in which verbs appear as an aid to meaning acquisition.[2] In the first part of this paper we use our program to examine the difficulty of using two particular syntactic constructions to discover the availability of non-stative senses. We conclude that only very modest syntactic and lexical capability are needed to exploit observation of these syntactic constructions for lexical semantic classification of verbs. This conclusion bolsters the general position of Gleitman (1989) that children can exploit the syntactic structure to aid in semantic classification of verbs.[3]

The second part of this paper describes current work on expanding the scope of the acquisition program both in terms of the lexical classifications it can learn and the quantity and type of text it can learn from. One of several benefits of this extension will be the ability to apply our learning program to corpora of maternal speech, thereby bringing it closer the child acquisition questions.

We have argued for the pursuit of automatic lexical classification based on syntactic context as it relates to questions about children's lexical acquisition, but we would also like to motivate it as technology.

There is wide agreement that language users, whether natural or artificial, need detailed semantic and syntactic classifications of words. However, most current approaches to satisfying that need for artificial devices do not involve learning from examples. Interpreting the information published in machine-readable dictionaries (e.g. Boguraev and Briscoe, 1987), entering it manually in conjunction with knowledge-base construction (Knight, 1989), and studying word collocations statistically (Church and Hanks, 1990) are alternative pro-

[1] The text source we've used so far is the Lancaster/Oslo/Bergen (LOB) Corpus, a balanced corpus of one million words of British English. All but a small percentage is edited prose.

[2] See Pinker (1984) and Pinker (1989) for a different perspective how child learners use constraints between lexical syntax and lexical semantics.

[3] We do not intend to claim that child learners exploit our particular syntactic constructions, which merely represent our earliest attempts; instead, we mean to model the general process of meaning classification based on the syntactic environments in which verbs appear.

posals. When learning from examples is proposed, it is usually tutored learning in a controlled environment (Zernik and Dyer, 1987). Ultimately, however, any language user must be able to add new words to its lexicon, if only to accommodate the many neologisms it will encounter. Moreover, researchers rarely agree on the necessary lexical classifications, and our lexicographic needs grow with our understanding of language. Any method that requires explicit human intervention — be it that of lexicographers, knowledge engineers, or "tutors" — will lag behind both the growth of vocabulary and the growth of linguistics, as well as being subject to the uncertainties of introspection. Further, the cost of maintaining dictionaries manually in the face of this growth will remain high. By contrast, dictionaries constructed by automated learners from real sentences will not lag behind vocabulary growth; examples of current language use are free and nearly infinite. And the ability of such dictionaries to keep pace with theoretical developments is limited only by the difficulty of coming up with syntactic tests and programming the system to detect revealing sentences. Judging by our experience so far, the former task will be the more challenging one.

2 Detecting Verbs with Non-Stative Senses

In this section we discuss our study of two syntactic constructions that reveal the availability of non-stative senses for verbs. This work focuses on three questions to determine the difficulty of discovering the availability of non-stative senses:

1. Is it possible to robustly detect sentences of the type illustrated in (1) and (2) using only a simple syntactic parse tree? How simple can the parse tree be?

2. Do the progressive and rate adverb tests actually behave as advertised in text, which is subject to performance limitations? Specifically, are the syntax-semantics constraints regular enough to support semantic classification under a broad range of psychologically plausible learning strategies?

3. Can a parse tree sufficient for Item 1 be reliably, automatically recovered using only a simple parser, a relatively compact grammar description, and knowledge of the major syntactic categories of the words involved?

Section 2.1 describes the two syntactic constructions we have studied and demonstrates their relation to the semantic category in question. Sections 2.2, 2.3, and 2.4, respectively, answer the three questions in the affirmative. Section 2.5 briefly describes the mechanism and resources used by the parser behind our lexical semantic learning program.

2.1 Revealing Constructions

The distinction between stative and non-stative verbs has been a subject of interest in linguistics at least since Lakoff (1965). Giving a precise semantic characterization of statives is rather involved (see Dowty, 1979); but, roughly speaking, they are verbs that, when asserted at some time, are assumed by default to hold at all later times. Classic examples of stative verbs are *know*, *believe*, *desire*, and *love*. A number of syntactic tests have been proposed to distinguish between statives and non-statives (again see Dowty, 1979). For example, stative verbs cannot normally appear in the progressive, (1). In

(1) a. OK Jon is fixing his car
b. * Jon is knowing calculus

addition, statives cannot be modified by rate adverbs such as *quickly* and *slowly*, (2). We have chosen the stative/non-stative distinction, and in particular the

(2) a. OK Jon fixes his car quickly
b. * Jon knows calculus quickly

two tests shown (1) and (2), as test cases for our approach to learning lexical semantic classifications from the syntax of example sentences.

2.2 Required Precision of Parse Trees

Consider first how much syntactic structure is needed to detect the progressive and rate adverb constructions. To begin with, let us assume that the availability of a non-stative sense is an intrinsic property of a verb independent of factors such as the subcategorization frame in which it appears.[4] To detect progressives one need only parse the auxiliary system. Rate adverbs, by contrast, require determining what the adverb modifies. For example, adverbs may appear after the direct object, (3a), and this must not be confused with the case where they appear after the subject of an embedded clause, (3b).

(3) a. Jon fixed the robot quickly
b. Jon knew his hostess rapidly lost interest in such things

Thus it is necessary to determine the boundaries of NPs and Ss. However, it is not necessary to know much about the internal structure of these phrases.

[4] This is a reasonable approximation for the stative/non-stative distinction. (The obvious exceptions are verbs like *think* that are stative with a propositional argument.) Subcategorization frames are essential for determining many other lexical semantic categorizations. See Section 3.

For example, it is not necessary to know the structure of noun-noun predications, or (except in contrived cases) the attachment of PPs. Finally, there are some truly ambiguous cases that cannot be resolved by any syntactic parser not already possessed of the distinctions we are attempting to learn, (4). When

(4) a. Jon fixed the robot that had spoken slowly
b. Jon believed the robot that had spoken slowly

encountering such sentences the strictly monotonic learner must recognize the ambiguity and decline to draw any conclusion. In summary, to the question: "Is it possible to detect sentences containing constructions (1) and (2) using only a simple syntactic parse?" we answer cautiously, "Yes, in principle."

2.3 Behavior of Test Constructions in the LOB

We now proceed to the question of whether the syntactic tests behave as advertised in real text, which is subject to the hazards of our linguistic performance as well as the rigors our competence. This question is addressed in two stages: first, we estimate the reliability of the progressive and rate-adverb constructions as indicators of non-stativity; next, we evaluate the implications of the reliability estimates for a broad range of psychologically plausible learning strategies. To investigate the reliability of our two constructions as indicators of non-stativity we processed the Lancaster/Oslo/Bergen (LOB) corpus of one million words of edited British English text. After partially parsing each sentence, our program automatically examined the parse trees and the words to see whether or not each clause in fact contained a progressive verb or a verb modified by a rate adverb. When the clause was determined to contain such a construction our program noted that in its dictionary entry for the appropriate verb. It also stored the example sentence for analysis by the researchers.[5]

Before considering the reliability estimates, note that we are exploring syntactic constructions that imply the availability of a **non-stative** sense for a verb, but not constructions that imply the availability of a **stative** sense. Accordingly, in conjecturing that a verb has a non-stative sense, we are quite concerned with false positives. Since we are not currently attempting to conjecture that verbs lack a non-stative sense there is no false negative case.[6]

If the progressive and rate-adverb tests behave ideally then every verb that shows up in one of these constructions ought to have a non-stative sense. To estimate how closely the test constructions approach that ideal, we examined by hand the observations our program recorded for the 100 verbs that occur

[5]Storing sentences is not, of course, required for the learning models under consideration here.

[6]After a thorough statistical analysis of our data we hope to explore the possibility of making negative conjectures stochastically. While such stochastic learning may prove valuable technologically, it is less relevant to the current focus.

most frequently in the LOB corpus. These 100 verbs occur about 50,000 times in the corpus, accounting for 50% of all verb occurrences. Of these 100 verbs, 89 occurred in either progressive or rate-adverb constructions at least once, according to our parser. Of the 89 appearing in the test constructions, only one verb, *mean*, lacks a non-stative sense. Of the 100 verbs, a total of six lack non-stative senses: *know, seem, mean, like, believe,* and *understand.* The lone false positive, *mean*, appears in a non-stative construction only once, in the anomalous sentence "It's a stroke, that was what he was meaning." The six verbs lacking non-stative senses appeared 3,835 times in the corpus with only this single occurrence in the progressive and no occurrences modified by rate adverbs.

Now we must translate the estimates of the reliability of the syntactic tests into conclusions about the reliability of various learning strategies that might employ them. The most obvious learning strategy, and the only one we have implemented so far, is to conclude that a verb has a non-stative sense immediately and irrevocably as soon as one of the telltale constructions is seen, independent of what has been seen in the past. This strategy requires no resources on the part of the learner. In particular, it requires no storage or counting of examples and no inference of any kind. Using this strategy, a learner exposed to our one-million word corpus would have mis-classified one in 89 of our verbs as non-stative, given six opportunities. However, it is obviously unrealistic to apply such a learning strategy over an unbounded body of examples — eventually any verb will show up in an anomalous context. Indeed, it is well-known that children retract overgeneralizations at many points in their language development (Brown, 1973). Nonetheless, the reliability of the syntax/semantics correlation in our sample was so strong that even the null strategy provided nearly one-hundred percent accuracy. Given only modest, psychologically plausible strategies and resources a learner could be expected to achieve one-hundred percent accuracy on the sample data. Although a more articulated and realistic learning strategy is beyond the scope of this paper, we hope to develop one shortly.

2.4 The Sufficiency of the Parser

In Section 2.2 we discussed how much structure must be imposed on sentences if the progressive and rate-adverb constructions are to be detected. In Section 2.3 we determined that the progressive and rate-adverb constructions are indeed reliable indicators of the availability of a non-stative sense. In this section we discuss the accuracy with which we can recover the parses deemed necessary in Section 2.2.

The question at hand is whether a sufficient parse tree can be reliably, automatically recovered using only a simple parser, a relatively compact grammar description, and knowledge of the major syntactic categories of the words involved. As mentioned above, we automatically parsed the LOB corpus. More

precisely, we parsed each sentence to the extent necessary to determine whether or not each clause in fact contained a progressive verb or a rate adverb. Let us consider the accuracy of our parser/analyzer in terms of the all-important measure of false positives. In other words, let us consider how many verbs that were judged to be either progressive or modified by a rate adverb in fact were not. It is not practical to check manually every verb occurrence that our program judged to be progressive. Instead, we checked 300 such sentences selected at random from among the most commonly occurring verbs. This check revealed only one sentence that was not truly progressive. That sentence is shown in (5a). Recognizing pseudo-cleft constructions like (5a) would require some

(5) a. *go*: What that means in this case **is going** back to the war years...
b. *see*: The task was solely **to see how speedily** it could be met...
c. *compare*: ...the purchasing power of the underdeveloped countries in the commonwealth **will rise slowly compared** with that of Europe.

thought and might entail additions to our grammar description or tree-analyzer that go beyond the merely cosmetic.[7] By contrast with the progressives, rate adverbs are infrequent enough that we were able to verify manually all 281 cases our program found. In four of those cases the rate adverb actually modified a verb other than the one that our program chose. Three of these four cases had the structure of (5b), where a wh- relative is not recognized as signaling the beginning of a new clause. This reflects an oversight in our grammar description that can be corrected trivially. The one remaining case of a mis-attributed rate adverb, (5c), would again require some attention to correct. The rate of false positives in sentence detection, then can be estimated at about one serious hazard in 300 for both tests. However, none of these wrongly identified sentences occurred with verbs lacking a non-stative sense, so none resulted in false positives in word classification.

2.5 Mechanism of the Parser

Having discussed the sufficiency of our parser for the task at hand, it is worth mentioning briefly the nature of the parser. It is a local, heuristic parser developed by DeMarcken (1990). De Marcken's parser is local in that its decisions about the boundaries of phrases are independent of its decisions about their attachment. For example, a grammar writer might choose to build PPs but not attach them. Whether or not they are attached does not affect the legitimacy of phrases or their boundaries. De Marcken's parser is heuristic in that the descriptions of legal structures and the decisions about where to look for those structures are decoupled. In other words, if the only NPs that are

[7] A gross but effective solution would be merely to ignore all sentences beginning with wh-words. This approach would simplify the subcategorization problem as well.

relevant are those that follow verbs and prepositions then the grammar writer can describe the structure of NPs in general, but attempt to build that structure only after a verb or preposition. The fact that the subject NP would then remain unparsed poses no problem for the parsing algorithm.

Beyond the parsing engine, the major resources are the parser rules and the lexical sources. The approximately one-hundred[8] parser rules we used are a subset of a fairly large grammar written by DeMarcken (1990). The subset was selected with the aim of meeting the requirements for structure described in Section 2.2. There is no reason not to use a more complete grammar other than the goal of demonstrating how little syntactic knowledge is needed to use the progressive and rate-adverb tests. The lexical syntactic categories used by the parser are a subset of those defined by the tagged LOB corpus. These include about ten open categories, roughly the ones found in any collegiate dictionary.[9] Even the most basic subcategorization information, such as whether a verb is transitive or intransitive, is unavailable.

In addition to the parsing engine, parsing rules, and dictionary, we used a variation on the DeRose (1988) statistical disambiguator (described in DeMarcken, 1990) to deal with lexical ambiguity. Fortunately, the LOB corpus is also available in a tagged form, where each word has been manually disambiguated. In order to check the effects of errors in statistical disambiguation on our conclusions we compared our results using the disambiguator to the results we got using the hand-tagged corpus. Specifically, we compared the number of progressive and rate-adverb constructions found for each of the verbs in the corpus. To our great surprise, the results were absolutely identical whether we used the disambiguator or the hand-tagged corpus. Although our disambiguator is fairly accurate, this identical performance on 100,000 verb occurrences dramatically demonstrates how insensitive the identification of these constructions is to lexical syntactic error. This one observation is perhaps the most impressive of all our results so far in demonstrating the ease of using these tests under the adverse circumstances with which children are faced.

3 Scaling Up

In order to realize the promise of the approach we have taken to semantic classification we must scale it up beyond the demonstration described above. Indeed, we are currently scaling it up both in terms of the quantity and type of text it can learn from and in terms of the lexical semantic classes it can learn. The ability to deal with text from diverse sources and in diverse formats

[8] It is impossible to convey precisely the power of the rules. They are more compact than context-free rules, but they seem not to be greatly so.

[9] Nominally, the LOB uses 132 categories, but most of these are either closed categories (such as all inflections of *do*, *be*, etc.) or inflections of open categories. We ignore the inflections, using our own morphological recognizer instead.

will have both psychological and technological benefits. On the psychological side, we hope to process corpora of transcribed maternal speech as well as the written text we have used so far. That will allow us to model much more accurately the challenges and opportunities that children would face in doing semantic classification from syntactic context. On the technological side, the log-normal frequency distribution of words in text[10] requires us to process very large quantities of text in order to compile a substantial lexicon. The sources of text that are available to us on the hundred-million words scale are completely raw, so a great deal of work is needed merely to divide the text into sentences. In addition we need sources for the major categories of the words, and these will be unavailable for many words, especially proper names. Indeed, even identifying the boundaries of proper names requires work. Despite the technological hurdles, we hope to be learning from newspaper text sometime this year.

The other way in which we must scale up is in terms of the number and diversity of semantic categories in which we can classify verbs. We are currently working on telicity, the distinction between processes and accomplishments/achievements (Brent, 1989). Work is also underway on the distinction between stage-level and object-level stative predicates (Dowty, 1979). Many of the semantic categorizations that we hope to learn, including telicity, are properties of the subcategorization frame in which a verb appears as well as of the verb itself. This means that our parser needs to be able to determine subcategorization frames. Although it currently goes a long way toward doing so, several hurdles remain. Nonetheless, as our ability to determine subcategorization improves we hope to be able to classify verbs of locomotion by their strong tendency to occur with directional prepositions and no direct object. Indeed, there are many classifications that can be learned using this type of subcategorization information. By expanding in this direction, we hope to make contact with the data on child language acquisition collected by Gleitman (1989), Pinker (1989), and others.

4 Conclusions

The data presented in this paper suggest that, with very modest syntactic and lexical capabilities and no semantic knowledge, it is possible to exploit the progressive and rate-adverb constructions to readily determine the availability of non-stative senses for many verbs. This conclusion is significant both for the study of child language acquisition and for the technology of automated, scientific lexicography. We do not claim that child learners exploit these particular constructions to do semantic classification. Rather, the psychological significance of this work lies in its demonstration that the syntactic contexts of verbs provide reliable information about their semantics which can be recovered with

[10] Zipf (1949)

minimal sophistication. The significance of our conclusion for scientific lexicography is its promise of permitting the automatic construction of large-scale lexica from primary sources. The automatic construction of such lexica is, in turn, one of the most promising paths in natural language processing technology.

References

[Boguraev and Briscoe, 1987] B. Boguraev and T. Briscoe. Large Lexicons for Natural Language Processing: Utilising the Grammar Coding System of LDOCE. *Comp. Ling.*, 13(3), 1987.

[Brent, 1989] M. Brent. Earning dividends on lexical knowledge: How the rich can get richer. In *Proceedings of the First Annual Workshop on Lexical Acquisition*. IJCAI, 1989.

[Brown, 1973] R. Brown. *A First Language: The Early Stages*. Harvard University Press, Cambridge, MA, 1973.

[Church and Hanks, 1990] K. Church and P. Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16, 1990.

[DeMarcken, 1990] C. DeMarcken. Parsing the LOB Corpus. In *Proceedings of the Association for Computational Linguistics*. Assocation for Computational Linguistics, 1990.

[DeRose, 1988] S. DeRose. Grammatical category disambiguation by statistical optimization. *Comp. Ling.*, 14(1), 1988.

[Dowty, 1979] D. Dowty. *Word Meaning and Montague Grammar*. Synthese Language Library. D. Reidel, Boston, 1979.

[Gleitman, 1989] L. Gleitman. The structural sources of verb meanings. Keynote address at Child Language Conference, 1989.

[Knight, 1989] K. Knight. Integrating language acquisition and knowledge acquisition. In *Proceedings of the First Annual Workshop on Lexical Acquisition*. IJCAI, 1989.

[Lakoff, 1965] G. Lakoff. *On the Nature of Syntactic Irregularity*. PhD thesis, Indiana University, 1965. Published by Holt, Rinhard, and Winston as *Irregularity in Syntax*, 1970.

[Landau and Gleitman, 1985] B. Landau and L. Gleitman. *Language and Experience*. Harvard University Press, Cambridge, MA, 1985.

[Pinker, 1984] S. Pinker. *Language Learnability and Language Development*. Harvard University Press, Cambridge, MA, 1984.

[Pinker, 1989] S. Pinker. *Learnability and Cognition: The Acquisition of Argument Structure*. MIT Press, Cambridge, MA, 1989.

[Zernik and Dyer, 1987] U. Zernik and M. Dyer. The self-extending phrasal lexicon. *Comp. Ling.*, 13(3), 1987.

[Zipf, 1949] G. Zipf. *Human Behavior and the Principle of Least Effort*. Addison-Wesley, New York, NY, 1949.

Sense Generation
or
How to Make the Mental Lexicon Flexible

Bradley Franks Nick Braisby

Centre for Cognitive Science
University of Edinburgh
2 Buccleuch Place
Edinburgh EH8 9LW
UK

Abstract

In this paper we address some key issues in the psychology of word meaning, and thereby motivate a *Sense Generation* approach to the diversity of senses that a word may have. We note that an adequate account must allow for the flexibility and specificity of senses, and must also make appropriate distinctions between default and non-default senses of a word, and between different senses for vague and ambiguous words. We then discuss two central components of a theory of sense. Firstly, *lexons*, the stable representations, in a "mental lexicon", of word meanings; secondly, *senses*, the mentally represented descriptions associated with particular *uses* of words. We argue that the crucial issues in accounting for the diversity of sense, are: the number of lexons we need to postulate, and the relationship between the contents of those lexons and their associated senses. *Sense Selection* accounts, of which we distinguish Strong and Weak versions, both of which find considerable support in the cognitive science literature, fail to account for the flexibility and specificity of senses in a way that is consonant with linguistic evidence regarding the ambiguity of words, and psychological evidence regarding the coherence which underlies their use. We will show how the *Sense Generation* approach, by positing a nonmonotonic relationship between lexons and their senses, respects these considerations. We sketch this approach, and finally note some of its promising implications for other aspects of word meaning.

1 Introduction

In this paper, our aim is to consider some possibilities for certain aspects of a theory of word meaning, and thereby to motivate what we call *Sense Generation*. We briefly outline the crucial phenomena of flexibility and specificity of senses in Section 2. In Section 3 we sketch some of the sorts of object which might be required in a theory of sense, and, most importantly, the relations between them. This provides the apparatus for a discussion of some possible theories of senses, in Section 4. In Section 5, we turn to a more detailed exposition of Sense Generation, indicating how it may account for flexibility and specificity. Finally, we sketch some implications of this view and touch on some wider concerns for theories of word meaning.

2 Preliminaries

The phenomena of flexibility and specificity are best illustrated by example.

> Mary is giving a dinner party at her home in the country. Unfortunately, her cupboards are bare. The appetites of her voracious guests are, however, whetted by the sight of Mary's pet mouse, Midge, tucking into some mouse food, and by the sight of Mary's pet canary eating some bird food. Aware of all this food-eating, one of Mary's less subtle guests asks "Do you have any food, Mary?". Mary replies that there is none but proceeds to feed her dog, Mungo.

How are we to resolve the semantical nature of the guests' problems? They are led to believe that Mary has no food and yet they can clearly see that she has: that she has pet food, but no food fit for human consumption. We require that any theory of word meaning respect the intuition that there are different but related senses attached to the word *food*, senses, for example, which apply to different types of food: human food, dog food, bird food, etc. We employ "sense" in a similar manner to Clark (1983): as the mentally represented aspects of the semantic content of a word on a particular occasion of use; we will be more precise about this in 3. The fact that the same word can seemingly have many different senses, illustrates what we call *diversity*. One aspect of the diversity of senses is illustrated by the fact that *food* seems to have senses corresponding to both types and subtypes of food. In this case, it has senses ordinarily associated with *pet food*, and its subtypes, for example *mouse food*. This aspect of diversity is what we call *specificity*: some senses of a word appear to be more specific than others. The fact that *food* may have senses for different types, for example, senses for "mouse food", "dog food", "mouse food", etc., illustrates another aspect of diversity, *flexibility*.

Our discussion of theories of sense will concentrate on several factors: we will be concerned to respect the arguments of, among others, Clark (1983) and Murphy & Medin (1985), which we will outline in more detail

later. We also require that any plausible account of reflect two important distinctions between types of senses: between senses that express default information and those that express non-default information, and between senses of vague and ambiguous words. Throughout, our overriding concern is to provide an evaluation which not only respects basic linguistic intuitions, but does so in a way that is consonant with a broad range of psychological considerations.

3 Some Components of a Theory of Sense

In this section, we will set out some aspects of a theory of sense. Two categories of object play a central role in our discussion. The first, *Senses*, are descriptions that we take to mediate relations between uses of words and their referents. These descriptions are both publicly specifiable and mentally representable. The notion of sense as employed here, although derived from that of Frege, does not carry a commitment to Frege's abstract semantical "third realm", distinct from the realms of mental and physical objects. The most important aspect of senses for our purposes is the way in which they guide linguistic behaviour. The application of a word to an entity (objects, events, substances - any individuum) is mediated by the sense of that word: in particular, the description that constitutes the sense subsumes the description of the object. So the uses of a word must be explicable in terms of the sense or senses which that word possesses. In this way, senses may be taken to classify the linguistic behaviour of agents.

The description of the phenomena of flexibility and specificity relied upon the various senses noted for *food* being different. This assumption was motivated by the application of what Evans (1976) labels the "intuitive criterion of identity" for senses. This determines that if a rational agent can both assent to and remain agnostic about the application of a referring expression to an entity when used in utterances of the same sentence, then that referring expression must have two different senses. As an illustration, reconsider Mary's dinner guests. Here, *food* is being used in different ways: sometimes it is being used to refer to all food, and at other times to types of food. So it is possible that one of Mary's guests could both assent to, and dissent from the statement, "Mary has food in her house". So it is quite felicitous for Mary to say, of the same entity (i.e., some mouse food, say), both that it is food, and that it is not. The intuitive criterion of identity for senses then requires that we treat *food* as having such different senses. Different senses express the fact that an entity may have different "modes of presentation" with respect to an agent: different ways the agent may refer to that entity. They also correspond to different ways of cognizing that entity: they are indicative of different perspectives that an agent may adopt.

The second type of object that we require is *lexons*. Most accounts of the psychology of language presuppose the existence of a "mental lexicon", in which words have "entries", that contain orthographic, phonological, morphological, syntactic and semantic information. The semantic component has been variously referred to as a "concept" or "lexical concept"; in order to avoid correlative unwarranted assumptions, we will refer to it as the lexon. A lexon, then, is a description that defines the stable mental representation in this mental lexicon; it also forms the semantic contribution of a word to the meaning of the expressions of which it forms a part. We also assume that senses are derivative in some way on lexons. That is, language users arrive at a sense for a word through first accessing its lexon. Given the multiplicity of senses which a word may have it is clear that a major problem for theories of is the relation between senses and lexons. Our discussion of such theories rests primarily on the way this issue is addressed.

Pre-theoretically, we are led to believe that senses usually outnumber the words with which they are associated: that is, the senses of a given word always number one or more. Considering the relations between words and their senses in terms of lexons then gives rise to two crucial questions. One concerns the *number* of lexons we postulate in order to effect these relations; and the other concerns the relations between the *contents* of senses and lexons.

In order to facilitate our discussion we will describe the contents of lexons and senses in terms of feature-structures like the following, which may describe the lexon for *chair*:

$$\begin{bmatrix} \text{legs:} & \begin{bmatrix} \text{number:} & 4 \end{bmatrix} \\ \text{seat:} & \begin{bmatrix} \text{number:} & 1 \end{bmatrix} \\ \text{made-of:} & \text{wood} \end{bmatrix} \quad (1)$$

This feature-structure is not intended to be an exhaustive specification of the content of the lexon for *chair*; it is presented for illustrative purposes only. If any lexon or sense has the same feature-structure as this one, we may conclude that they are in fact identical lexons/senses. There may be cases where one feature-structure subsumes another, by having the same content or some addition of features. Another possible case is where two feature-structures cannot be ordered by this relation. The former is indicated by the relationship between the structures for *chair* and *arm-chair*, and the latter by that between *chair* and *rocking chair*:

$$\begin{bmatrix} \text{legs:} & \begin{bmatrix} \text{number:} & 4 \end{bmatrix} \\ \text{seat} & \begin{bmatrix} \text{number:} & 1 \end{bmatrix} \\ \text{arms:} & \begin{bmatrix} \text{number:} & 2 \end{bmatrix} \\ \text{made-of:} & \text{wood} \end{bmatrix} \quad \begin{bmatrix} \text{legs:} & \begin{bmatrix} \text{number:} & 0 \end{bmatrix} \\ \text{seat} & \begin{bmatrix} \text{number:} & 1 \end{bmatrix} \\ \text{arms:} & \begin{bmatrix} \text{number:} & 2 \end{bmatrix} \\ \text{rocker:} & \begin{bmatrix} \text{number:} & 2 \end{bmatrix} \\ \text{made-of:} & \text{wood} \end{bmatrix} \quad (2)$$

The subsumption relation amounts to the kind of relation that holds between a type and one of its tokens; typically, we might assume that that defining features of a type are possessed by a token that can be categorised as a member of that type.

On just these two dimensions, the number and contents of lexons and senses, we distinguish three classes of theory. The first, Strong Sense Selection ($\mathcal{S}$), is that which Clark demonstrated to be unsound; we will note some additional problems. $\mathcal{S}$ may appear to be a straw man; a more plausible alternative is the second class, Weak Sense Selection ($\mathcal{W}$). $\mathcal{W}$ has two variants, both of which appear to be flawed. The third class, Sense Generation, avoids these difficulties, and is the one we would like to endorse.

4 Problematic Theories of Sense

4.1 Strong Sense Selection

Two assumptions identify $\mathcal{S}$. Firstly, the number of lexons: there is a lexon for each and every sense of a given word. Secondly, the contents of the lexons and senses: $\mathcal{S}$ assumes that the contents of each lexon and its corresponding sense are identical. $\mathcal{S}$ accords well with standard model-theoretic analyses of word meaning. For example, the approach taken in Montague-style semantics requires that different interpretations for the same syntactically unambiguous linguistic string result from the same word having different basic expressions. In the case of *bank*, for example, there would be two distinct basic expressions, $bank'_1$ and $bank'_2$, in the lexicon. $\mathcal{S}$ then offers the possibility of being able to treat the diversity of senses which might be associated with *food* in the same way, and thus to provide a precise semantics. All of the idiosyncratic information which demarcates senses is thus represented in lexons.

The diversity of senses that might be attached to *mother* provides another illustration of $\mathcal{S}$. Such senses include "adoptive mother", "biological mother", "surrogate mother", "foster mother" and "step-mother". According to $\mathcal{S}$ each of these senses is assigned a distinct lexon whose content expresses that of the sense.

Despite its prevalence in formal approaches to word meaning, $\mathcal{S}$ has some irremediable deficits. Some of these, relating to flexibility, coherence and ambiguity, are also problems of $\mathcal{W}$ and we will turn to these in 4.3.

However, there are also problems unique to $\mathcal{S}$. One concerns the number of lexons we are led to hypothesise in order to capture diversity. This is essentially the same point that Clark made in respect of "nonce" senses; however, as we have seen, even for common nouns such as *food* and *mother* the number of senses greatly exceeds the number of words. Accordingly, in $\mathcal{S}$, so does the number of lexons. This is problematic since the multiplication of lexons must make psychological sense. It is unclear that this is so in the case of $\mathcal{S}$ since such a multiplication places an intolerable burden on memory and presumably would result in a highly complex search procedure. These problems are difficult enough in the case of the interpretation of single words: in the case of combinations, such as simple noun phrases, there would be an explosion of combinatorial possibilities, in which the appropriate sense would have to be selected from a list comprising each and every permutation of all of the lexons associated with each constituent.

Regarding flexibility, $\mathcal{S}$ appears to proffer a solution that treats vagueness and context-sensitivity in the same way as ambiguity: by postulating independent lexons for each sense of a vague word. We will return to this in 4.3. It is also clear that specificity raises difficulties for $\mathcal{S}$. The issue is whether we can have a limitless number of ever more specific senses for a given word. We remain agnostic about this possibility though it is clear that very many senses may be associated with the same word. However, $\mathcal{S}$ rules out the possible unboundedness of specificity by *fiat*. That is, the only way that $\mathcal{S}$ can possibly capture specificity is via the multiplication of lexons, and, given the uncontroversial assumption of a finite lexicon, the possible unboundedness of specificity could not be captured by $\mathcal{S}$.

$\mathcal{S}$ also appears unable to distinguish between senses which express default information and those which express non-default information. For instance, the default sense for *mother* is, presumably, "biological mother" yet this sense is accorded the same status as the other senses of *mother*. That is, they are each assigned a separate lexon. Of course, $\mathcal{S}$ theorists may have in mind some other bit of theoretical apparatus which to capture this distinction. The fact is, though, that as it stands, $\mathcal{S}$ does not respect this very important distinction. An alternative to this rather straw-mannish way of trying to capture the phenomena is offered by Weak Sense Selection.

4.2 Weak Sense Selection

$\mathcal{W}$ is characterised by three assumptions: firstly, there may be more than one lexon for a given word; secondly, there may be more than one sense for a particular lexon. Thirdly, it is assumed that the contents of senses and the contents of corresponding lexons lie in the relation of subsumption: that is, the only possible difference between a sense and the lexon from which it comes, is that the sense may have had features added. $\mathcal{W}$ is more appealing than $\mathcal{S}$ in the following ways. Firstly, senses seem to be intrinsically context sensitive: in conventional circumstances *mother* has the sense of "biological mother", but in a social work inquiry, for example, *mother* may have the sense of "biological mother who is also a carer". A way in which this context-sensitive specificity can be captured is through some process by which features are added to the contents of lexons in a manner appropriate to context. Though $\mathcal{W}$ does not specify such a process it is clearly implicit in its definition. This aspect of $\mathcal{W}$ is in the spirit of the findings of Barsalou (1982): different senses may be different "context-dependent" elaborations of a single "context-independent" lexon.

Another appeal of $\mathcal{W}$ is the fact that it allows lexons to express generalisations with respect to the category to which a word applies. That is, $\mathcal{W}$ allows that *lion* may have various senses but that the lexon for *lion* may be a description that applies to all (and only) lions. This is again quite appealing given standard assumptions about word meaning. The arguments of Kripke (1972) and Putnam (1975) for example, assume that senses apply to all and only those individuals to which the word applies. Further, the fact that $\mathcal{W}$ distinguishes between the different senses deriving from a lexon, that is, between those whose content is the same as the lexon and those whose content is an elaboration of the lexon's, *may* allow for the expression of default information. That is, default information might be expressed as part of the content of lexons.

There are two extreme versions of the $\mathcal{W}$ thesis: one is that the number of senses and lexons are equal, which forces equivalence with $\mathcal{S}$; another is what we might term the "Generality" option. This results from the assumption that a word has only one lexon, whose content may be added to and made more specific. $\mathcal{W}$ allows that the number of lexons may be intermediate between the number of words and the number of senses: what we will term an "Intermediate" option. Our previous discussion of $\mathcal{S}$ allows us to consider just the Generality and the Intermediate options of $\mathcal{W}$. The Generality option would operate in the following way. *Mother*, for

example, would be assigned a single lexon whose content would subsume all the senses that *mother* can have. So all of its senses result from the addition of features to this lexon's content. Given the diversity of senses for *mother* such a lexon must needs be maximally unspecific. In contrast, the Intermediate option allows *mother* to have more than one lexon underlying its senses. For example, we might have lexons for "biological mother" and "surrogate mother", say. The latter might be further specified to yield senses for "adoptive mother", "step-mother" and so on.

There are a number of problems with $\mathcal{W}$ some of which we will deal with in 4.3. However, we will outline some problems unique to $\mathcal{W}$ here. A critical problem of the Generality option is that it appears to be unable to express default information. Reconsider the example of *mother*. The appropriate lexon cannot express a relation of genetic inheritance because although some mothers are related this way to their children and some are not (e.g., foster mothers). To specify such a feature in the lexon would be to exclude mothers such as these from the domain of application of the lexon and all its associated senses, since features can only be added to the lexon and not taken away. For similar reasons, the lexon cannot express any relation of caring between mothers and children. And so on for any other feature-specification which we might ascribe to the lexon for *mother*. Arguments such as these indicate that in many cases the Generality option leaves us with a maximally unspecific lexon. This, however, flies in the face of the strong intuition that words *do* have default senses. The fact that this option renders the expression of such defaults as difficult to obtain as the expression of exceptions is a major deficit.

The Intermediate option postulates a *certain* multiplicity of lexons: there might be more than one lexon for *mother* (in contrast to the Generality option), but less than would be postulated by $\mathcal{S}$. The critical problem here is exactly how the number of lexons might be determined. Whereas for $\mathcal{S}$ and the Generality option, there is an overt constraint on the number of lexons postulated, it is not clear what principle there could be for deciding on the number of lexons in Intermediate option. For example, what lexons might we postulate for *chair* so that it may have a sense corresponding to "rocking-chair"? Since feature-adding is the only way in which senses may be derived from lexons, the sense for "rocking-chair" shown in (2) can only be derived from a lexon which either does not specify the number of legs or specifies no legs. Under this Intermediate option we are allowed to postulate several lexons for *chair*: one corresponding to (1), say, and one corresponding to that for "rocking-chair" in (2). A problem may arise, however, in the case of a special type of rocking-chair having no legs and no rockers. Assuming that *chair* can have this sense, the question arises as to how it is derived from the lexons we have postulated. If we only have feature-adding at our disposal, such a sense simply cannot be derived from either of the lexons for chair we have postulated. Our only option is to suppose that there is another lexon for chair. The problem is that, in principle, there seems to be no bound to the number of exceptional chairs we can imagine and for each type, we would be led to posit an additional lexon. The issue then would be, what degree of exception do we rule out as invoking a new lexon? One way of constraining the number of lexons might be to determine a threshold for permitted specificity of lexons: if a particular sense is more specific than the threshold level, it must be represented as a sense deriving from a particular lexon (and not as a lexon in itself). Three problems render such a criterion untenable.

Firstly, it is not clear just how we could go about comparing the relative specificity of senses that have non-overlapping contents; for example, is "biological mother" less specific than "adoptive mother"? The operation of this criterion is perspicuous within groups of senses that *can* be ordered according to specificity (i.e., where the only difference between senses is in the degree of specification of the same set of features), but not in groups that cannot be so ordered. The general application of such a criterion would require a complete theory of content for lexons and senses, and some precise and motivated means for comparison. Neither are at present available. Secondly, to stipulate that lexons must be relatively non-specific may mean that they cannot express default information, since this is typically quite detailed and specific in nature. A third problem concerns the plausibility of postulating independent lexons to account for senses that are discriminable though related. This will be picked up in the next section.

4.3 Difficulties with Sense Selection

There are three principal flaws common to Sense Selection accounts. The first concerns the multiplication of lexons. The second concerns the ability of Sense Selection to account for the full range of flexibility. And the third concerns the underlying commitment to monotonicity, which gives rise to the first two problems.

The difficulty with assuming multiple lexons is that it is not clear to what extent they plausibly reflect mental representations. Arguments from linguistics and psychology caution against unprincipled multiplication. The linguistic considerations concern the difference between ambiguity and vagueness. Accounting for the diversity of senses by postulating distinct underlying lexons assumes that they are, synchronically, wholly independent. That is, supposing different lexons for a given word assumes they are as different as different lexemes with a single orthographic/phonological form. Postulating two lexons for *food* ("animal food" and "human food") treats *food* as an ambiguous item like *bank*. This amounts to making no distinction between different senses of a vague term, and different senses of an ambiguous term. However, if we consider any of the standard linguistic tests for ambiguity (Cruse, 1986), then we find that the independence of content assumed by postulating different lexons does not hold for examples such as *food*. For example, consider zeugmatic contexts (those which give rise to two different senses of a word at one time): "He sat on the bank whilst fishing and put his cheque in it". The strong contrast or opposition between the two senses of *bank* requires the postulation of independent lexons to account for them. In contrast, the various senses for *food* or *mother*, noted earlier, do not produce an opposition of sufficiently marked character. The examples are, rather, characterised by the relatedness of the various senses: they are distinct but clearly *not* independent.

Multiplication of lexons is also countered by psychological considerations raised by Murphy & Medin (1985). Murphy & Medin's discussion bears on the issue of the mental representations underlying the application of words to referents. In the current framework, these are lexons. The question for both $\mathcal{W}$ and $\mathcal{S}$ is whether the postulation of multiple lexons accords with psychological evidence. Murphy & Medin's arguments convince us that such multiplication is unwarranted. Consider whether the postulation of independent lexons for "animal food", and "human food", say, is justified on psychological grounds. The thrust of Murphy & Medin's arguments is to suggest that categories such as food are highly structured, and that the application of *food* to individuals thereof is highly dependent upon this structure. That is, the application of a word to entities in a category reveals what Murphy & Medin call "coherence". And the crucial point regarding coherence is that the application of a word to such an entity is dependent upon our theories concerning that entity. It is the fact that entities can be related by theories that allows their grouping together to be psychologically plausible. If lexons are to be psychologically plausible, then entities that form a coherent category (e.g., all different types of mother) should all fall under the extension of the same lexon.

However, even if we were to allow some multiplication of lexons - that is, even if the above considerations have no purchase - there are still cases in which the postulation of independent lexons to underly senses for a particular word would be implausible. These include the "contextual expressions", discussed by Clark (1983), and many examples noted by Nunberg (1977). Nunberg notes several different possible referents (and therefore, senses) for *newspaper*: a particular token of the newspaper (as in, "here's your newspaper, sir!"), the newspaper company as a whole (as in, "the newspaper's profits are less than expected!"), and a particular journalist (as in the case of a dubious piece of governmental behaviour: "don't say a word, the newspaper is here!"). It is clear that we would not want to claim that *newspaper*, for example, has a pre-stored lexon that expresses the sense of the third use ("journalist"). Yet this is precisely what Sense Selection accounts would have to hypothesise, since this sense of *newspaper* could not be said to be a simple specification of a lexon for *newspaper* (as $\mathcal{W}$ might aver). Clark argues that the parsing of contextual expressions (including certain denominal verbs, such as *to teapot*) stems from the *creation* of interpretations associated with those phrases. It is clear that the same kinds of considerations apply equally to Nunberg's examples. Since the only possibility of a creative process for Sense Selection is provided by the specification mechanism of $\mathcal{W}$, the only contextual expressions that $\mathcal{W}$ could accommodate would be those that are mere specifications of pre-stored lexons. And this cannot capture the flexibility evidenced in, for example, denominal verbs. It is clear that, even though Clark's discussion of $\mathcal{S}$ might appear to have attacked a straw man, his arguments have a broader significance, and have played a central role in undermining the more plausible $\mathcal{W}$.

The preceding discussion leads to the conclusion that none of the versions of Sense Selection are adequate to the task of accounting for the phenomena in a way that does justice to basic psychological and linguistic intuitions and requirements. The major problems stem from Sense Selection's adherence to monotonicity. That is, to the assumption that any alteration in the content of a lexon in the formation of a sense must be feature-addition. The Sense Generation approach (section 5) circumvents these problems by denying precisely this assumption and then tracing the ramifications.

5 Sense Generation

In opposition to the above types of theory, Sense Generation regards the variation we observe in senses to be due not to variation in lexons nor the generality of lexons but to some generative process which generates various senses from a (lexon) base.

5.1 Aspects of Sense Generation

Sense Generation is characterised by the following assumptions. Firstly, the number of lexons is identical to the number of non-ambiguous words. A single linguistic string is assigned more than one lexon if and only if it has genuinely unrelated senses, as indicated by tests for ambiguity. Secondly, the content of a lexon comprises the default sense of the word. Thirdly, the different senses of a word are generated from the lexon for that word. Fourthly, generation may result in a sense that is non-monotonically related to the lexon; that is, generation may result in a sense that does not simply add features to those of the lexon: features may be retracted or negated in the generation process.

Such a view can readily account for the kinds of example that are so problematic for Sense Selection. Recall the different senses for *newspaper*. In Sense Generation the lexon corresponds to the default sense of *newspaper*, perhaps as in "the newspaper hit the mat". The two other senses we have identified would then be non-monotonically derived from this lexon. The sense, for example, in "the newspaper's profits halved" would have to involve a retraction of those features expressing the facts that newspapers are material objects, made of paper, containing ink, etc. It would also need to include features expressing facts about businesses, finance, employment, etc. What Sense Generation claims is that this latter sense can indeed be generated from the lexon for *newspaper*. That is, there is some process by which features are negotiated. The precise nature of such processes is, clearly, a matter for further empirical enquiry but we suggest two possibilities. One is that argued for in Franks (1989) which involves the emergence of the features of a sense being constrained by some implicitly attached noun derived from an instantiation. For example, the sense of *fake gun* may be partly derived from the lexon for *replica* (thus adding features such as the degree of resemblance to a gun, and the way the object might be constructed), which is accessed as a result of the lexon for *fake* defeating certain features of that of *gun* (like firing bullets). Another is that suggested in Braisby (1989) where a related process of combining lexons results in the defeating of features. For example, the sense of *lion* which applies to stone lions, can be seen to result from the combination of a relational lexon such as "statue" and the lexon "lion". Similarly, for other non-default senses: they are derived from the combination of default and other lexons.

5.2 Implications of Sense Generation

The link between Sense Generation and Clark's *sense creation* should be noted. Clark's argument is that there is a restricted and well-defined group of contextual expressions, for which a sense creation process is necessary. Clark suggests that, for other kinds of expressions, a selection mechanism may be adequate. Since Clark's focus of attention is not conventional senses, the impression may be gained that these are not similarly contextual in nature. Where Sense Generation differs from sense creation is in its firm committment to the view that conventional uses are higly contextual and therefore require some generative process to explain the diversity of their senses. It follows that there is no clear dichotomy between contextual expressions and those used conventionally, rather there is a gradation.

Sense Generation is, in general, agnostic about the precise time-course of the role of *context* in determining senses. Indeed, it is meant as a formal, abstract characterisation of a class of theory. Consequently, it is compatible with more detailed accounts which suppose context to play a pre-access as opposed to a post-access role. That is, although context may choose from a number of possibilities for generation, it may also rule out certain possibilities prior to any generative process. Ultimately, this entails that exceptional senses need not require a longer time-course than default ones. Clearly, the exact time-course underlying the generation of senses is a matter for empirical enquiry: it suffices to note that Sense Generation is compatible with either outcome.

One implication of Sense Generation is that there are meanings of which senses are descriptions. Two questions

arise: is there a meaning relation which the default sense of, say, *mother* describes? And, since there are many senses of *mother*, are there also many meanings to *mother*? Whereas the traditional theory of meaning may answer these questions in the negative, support for affirmative answers comes from *Situation Theory*. Indeed the notion of meaning relations as conditional constraints allows us to claim that default senses are descriptions of meanings. The assumption that there are many meaning relations underlying the uses of a word is also perfectly compatible with the framework of Situation Theory.

A further range of issues concerns the connection between Sense Generation and various hypothesised structures and contents for lexons. The Sense Generation approach allows us to endorse certain aspects of both *classical* and *prototype* representations, whilst rejecting problematic implications of both views. In terms of the epistemological rationale of the classical approach, Sense Generation rejects the search for common features or necessary conditions underlying the sense of a word (as a result of nonmonotonicity), whilst it allows us to retain the economical representations that would result were the search for necessary conditions successful. In contrast, the relations between the various generated senses for a word mirror the intuitive and epistemological underpinnings of family resemblance (as a result of nonmonotonicity and the emergence of new features), which is acheived without postulating prototype representations, with their attendant difficulties. That is, the various senses generated for a word in different contexts will be related by a family resemblance, in line with Wittgenstein's (1953) original formulation.

Senses are descriptions that mediate a word's reference. There may appear to be a tension between this fact and the unlimited scope for nonmonotonicity in Sense Generation: if generation defeats all of the default features of a word's lexon, then we may refer to an entity through a description having nothing in common with the usual properties of the type of referent. This point is countered by considering the *perspectival-relativity of categorisation*. A situated agent, in referring to an entity through a particular sense, can be seen to be adopting a perspective on that entity. Even though Sense Generation allows of the logical possibility that words may refer to any manner of entity, an important constraint is deemed to operate. Namely, the agent in making such a reference must be adopting a particular perspective. While a theory of perspectives is something we lack, we note that for such a reference to be posited there must be independent evidence concerning the perspective adopted. Further, we suppose that the nature of the perspectives that people may adopt is such as to determine the content of senses which they relate to referents. This may of itself limit the degree of permissible difference between the content of a word's lexon and its senses, since one of the purposes of perspectives is to allow mutual reference. That is, one constraint is that several agents must be able to share a single perspective.

6 Conclusion

In this paper, we have sketched some of the assumptions that underly prevalent views of the senses of words (Sense Selection views). We have also considered some of the problems that arise when such views attempt to account for some basic phenomena of word meaning in a way that is consonant with linguistic and psychological desiderata. This then motivated an alternative view that circumvents these difficulties (Sense Generation). Finally, we noted some of the other advantageous implications of Sense Generation.

References

Braisby N. R. (1989). Situating Word Meaning. Paper presented at the Conference on Situation Theory and its Applications, Asilomar, California. March, 1989.

Clark, H. H. (1983). Making Sense of Nonce Sense. In d'Arcais, G. B. F. and Jarvella, R. J. (eds.). *The Process of Language Understanding*. Chichester: John Wiley and Sons. 297-331.

Evans G. (1976). *The Varieties of Reference*. Oxford: Basil Blackwell.

Franks B. W. (1989) Criteria and Concepts: an Anti-Realist approach to Word Meaning. PhD. Thesis. Centre for Cognitive Science, University of Edinburgh.

Murphy G. L. & Medin D. L. (1985). The Role of Theories in Conceptual Coherence. *Psych. Rev., 92,* 289-316.

A DISTRIBUTED FEATURE MAP MODEL OF THE LEXICON *

Risto Miikkulainen
Artificial Intelligence Laboratory
Computer Science Department
University of California, Los Angeles, CA 90024
risto@cs.ucla.edu

Abstract

DISLEX models the human lexical system at the level of physical structures, i.e. maps and pathways. It consists of a semantic memory and a number of modality-specific symbol memories, implemented as feature maps. Distributed representations for the word symbols and their meanings are stored on the maps, and linked with associative connections. The memory organization and the associations are formed in an unsupervised process, based on co-occurrence of the physical symbol and its meaning. DISLEX models processing of ambiguous words, i.e. homonyms and synonyms, and dyslexic errors in input and in production. Lesioning the system produces lexical deficits similar to human aphasia. DISLEX-1 is an AI implementation of the model, which can be used as the lexicon module in distributed natural language processing systems.

1 Introduction

The lexicon in symbolic NLP systems is a list of word symbols and phrasal patterns, with pointers to conceptual memory. The memory contains syntactic and semantic knowledge about the lexicon entry in the form of declarations, or procedures which specify how the word should be interpreted in different environments [29; 1; 6]. This knowledge has been explicitly programmed into the system with specific examples in mind. The symbolic lexicons are intended to model the *processes* of lexical access, not the physical structures that implement the processes. Consequently, these models lack the capacity to account for lexical errors in human performance, as well as lexical deficits in acquired aphasia.

A number of connectionist models of lexical disambiguation have been proposed [5; 25; 7; 12; 8]. These models aim at explaining lexical processing with low-level mechanisms, and can better account for the timing of the process, as well as for certain types of performance errors and deficits. However, they are still primarily process models, detached from the physical structures. They are designed as controlled demonstrations, not as building blocks in larger NLP systems.

*This research was supported in part by an ITA Foundation grant and by fellowships from the Academy of Finland, the Emil Aaltonen Foundation, the Foundation for the Advancement of Technology and the Alfred Kordelin Foundation (Finland).

The main goal of the DISLEX project (DIStributed feature map LEXicon) is to develop a computational model of the human lexical system, which is plausible at the level of *physical structures* such as maps and pathways. The model is based on current cognitive neuroscience theories and accounts for several documented lexical deficits in acquired aphasia and dyslexia. A secondary goal is to build a practical implementation of the model for a distributed story understanding system [19].

In terms of the symbolic lexicon models, DISLEX contains both the symbol memory and the conceptual memory, and implements a mapping between them. However, DISLEX is based on distributed representations of the word symbols and the word semantics. The lexical system is seen more like a filter, which transforms an input word symbol into its semantic representation, and vice versa. The memory organization and the mapping are formed in an unsupervised self-organizing process, based on examples of co-occurrence of the word and its meaning. As a model of the lexical system, DISLEX is in good agreement with Caramazza's theory [3]. The architecture offers a simple explanation to several types of lexical errors and deficits.

2 Overview of DISLEX

DISLEX has separate symbol memories for each input and output modality (figure 1). These memories store distributed representations for the physical word symbols, which are used in communication with the external world. For example, an orthographic word representation for `DOG` consists of the visual form of the letters `D`, `O`, `G`, while the phonological representation stands for the string of phonemes `do:g`. The separation of modality-specific channels is intuitively compelling, since the modalities give rise to different representations, and are processed through different structures [3]. The symbol spaces are not identical across modalities, there are homophones and homographs. Considerable experimental evidence also supports dissociation of the lexical components [3] (section 8).

The semantic memory of DISLEX consists of distributed representations of meanings, called semantic words. The semantic word **dog** (or e.g. **dog32**) refers to a specific animal and contains information such as domestic, mammal, brown color etc. There is a pathway from the semantic memory to the higher level language processing systems, which use semantic representations. The semantic memory

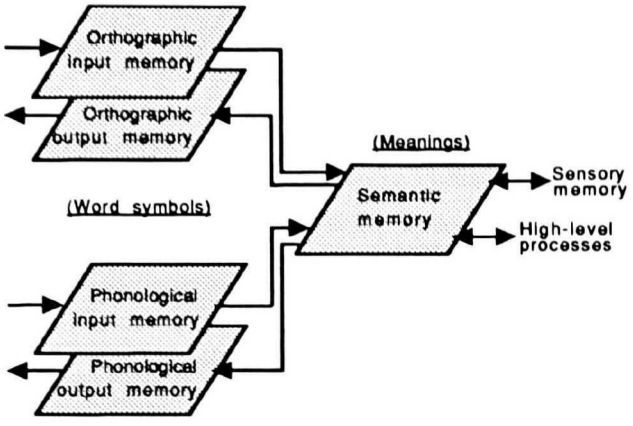


Figure 1: **The DISLEX architecture.** The physical symbol memories are modality and direction specific. The arrows indicate pathways of distributed representations.

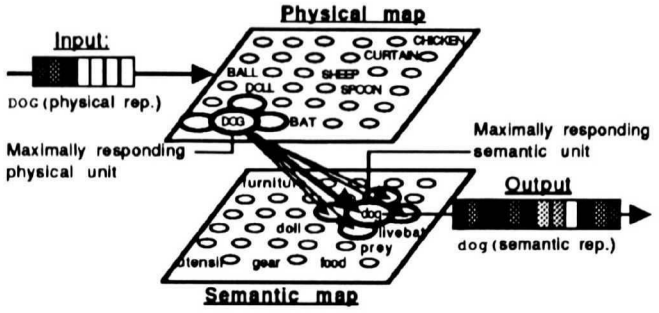


Figure 2: **Physical and semantic feature maps.** The physical input word DOG is transformed into the semantic representation of **dog**. The representations are vectors of real values between 0 and 1, shown by gray-scale coding. The size of the unit indicates the strength of its response. Only a few strongest associative connections are shown.

is also connected to the sensory memory, which contains visual images of objects and other sensory information. This pathway allows nonlinguistic access to the semantic memory, and provides the means for symbol grounding. The semantic word representation contains sensory information about the word referent, and the abstract word meaning originating from the high-level processes (ID and content, see [22]).

The physical and semantic memories are implemented as feature maps (figure 2). There is one map for each input and output modality and one for the semantic memory. The maps lay out each high-dimensional representation space on a 2-D area so that the similarities between words become visible. Physical words with similar form, e.g. **BALL, DOLL** are represented by nearby units in a physical map. In the semantic map, semantic words with similar content, e.g. **livebat, prey** are mapped near each other.

The physical maps are densely connected to the semantic map with associative connections. A localized activity pattern representing a symbol in the physical input map will cause a localized activity pattern to form in the semantic map, representing the meaning of the symbol (figure 2). Similarly, an active meaning activates a symbol in the physical output map. The lexicon thus transforms a physical input representation into a semantic output representation, and vice versa, and serves as an input/output filter for language processing. The physical and semantic maps are organized and the associative connections between them are formed simultaneously in an unsupervised learning process.

3 The DISLEX-1 simulation

DISLEX-1 is an AI implementation of DISLEX, designed as the lexicon module for a distributed neural network story understanding system [19]. DISLEX-1 contains a single physical modality, and the same representation space is used for both input and output. Figure 2 displays the basic architecture of DISLEX-1. Associative connections exist in both directions (the connections from semantic to physical map are omitted from the figure), and the transformation depicted in the figure can be reversed. This is a practical design for an AI module, and illustrates the basic principles and properties of the model.

DISLEX-1 was trained with data from a sentence processing experiment [17; 21] (figure 3). In the remainder of the paper, the mechanisms and properties of DISLEX are discussed, using the DISLEX-1 simulation as an example.

4 Representations

4.1 Physical representations

A central assumption in DISLEX is that the representations in each physical modality reflect the similarities within that modality. For example, the orthographic representations for DOG and DOC are very similar, but less so in the phonological domain.

The DISLEX-1 architecture concentrates on the orthographic modality. A simple encoding scheme was used to build the distributed representations for the written words. Each character was given a value between 0 and 1 according to its darkness, i.e. how many pixels are black in its bitmap representation. The darkness values of the word's characters were then concatenated into one representation vector (figure 3). This simple representation adequately reflects the visual similarities of the orthographic word symbols.

4.2 Semantic representations

The semantic representation is a distributed representation of the meaning of the word. Semantic representations are used internally for processing in cognitive models, and they should facilitate inferencing, expectations, generalizations etc. [15; 22]. A possible solution is to compose the representation from an ID part, representing the sensory referent of the word, and a content part, which encodes the processing properties of the word in relation to other words [22]

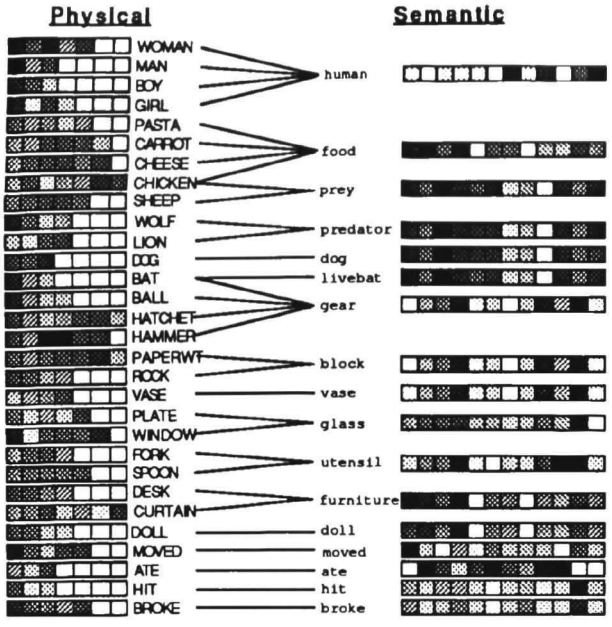


Figure 3: **The training data for DISLEX-1.** The physical representations code the orthographic word symbols, while the semantic representations stand for distinct meanings. Gray-scale boxes indicate component values within 0 and 1. The connections depict the mapping between the symbols and their meanings.

Category	Semantic words
animal	**prey predator livebat dog**
fragileobj	**glass vase**
breaker	**gear block**
hitter	**gear block vase**
possession	**gear vase doll dog**
object	**gear block vase glass food furniture doll utensil**
thing	**human animal object**
verb	**hit ate broke moved**

Table 1: **Semantic categories.** Each slot in the sentence templates specifies a category, and can be filled with any semantic word in that category. In other words, the categorization determines how the words are used in the sentences.

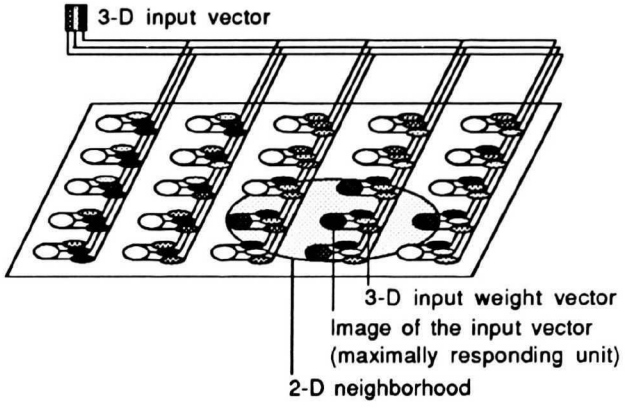


Figure 4: **A self-organizing feature map network.** A mapping is formed from a 3-dimensional input space onto a 2-dimensional network. The values of the input components, weights and the unit output are shown by gray-scale coding.

With the FGREP-mechanism [21] it is possible to extract the processing content of the word from examples of its use, and code it into a distributed representation. An FGREP-module is a three-layer backpropagation network which automatically developes distributed representations for its input items as it is learning a processing task.

For simplicity, and without restricting the generality of the model, the sensory part was omitted from the training data for DISLEX-1. The semantic representations for DISLEX-1 were formed with FGREP in the sentence case-role assignment task. The input to the FGREP network consisted of the syntactic constituents of the sentence and the network was trained to assign the correct semantic case roles to them. The sentences were generated from templates, by filling each slot in the template with a word from a specified category (table 1). The actual sentences and the specifics of the task are not important for this discussion (see [21]). However, the meanings embedded in the semantic representations originate from the categorization in table 1.

The representations that result from the FGREP process reflect the use of the semantic words (figure 3). Words belonging to the same category have a number of uses in common, and their representations become similar. The total usage is different for each word, and consequently, they stand for unique meanings.

5 Word maps

5.1 Topological feature maps

A 2-D topological feature map [13] implements a topology-preserving mapping from a high-dimensional input space onto a 2-D output space. The map consists of an array of processing units, each with N weight parameters (figure 4). The map takes an N-dimensional vector as its input, and produces a localized pattern of activity as its output. In other words, an input vector is mapped onto a location on the map.

Each processing unit receives the same input vector, and produces one output value. The response is proportional to the similarity of the input vector and the unit's weight vector. The unit with the largest output value constitutes the image of the input vector on the map. The weight vectors are ordered in such a way that the output activity smoothly decreases with the distance from the image unit, forming a localized response.

The ordering of the weight vectors retains the topology of the input space. This means roughly that nearby vectors in the input space are mapped onto nearby units in the map. This is a very useful property, since the complex similarity relationships of the high-dimensional input space become visible on the map.

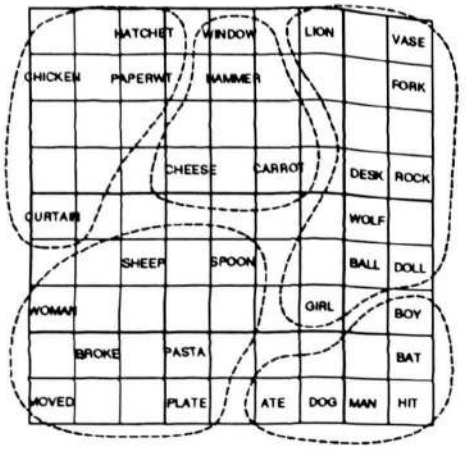


Figure 5: **The physical map.** Each unit in the 9×9 network is represented by a box in the figure. The labels indicate the image unit for each physical word representation. The map is divided into major subareas according to word length.

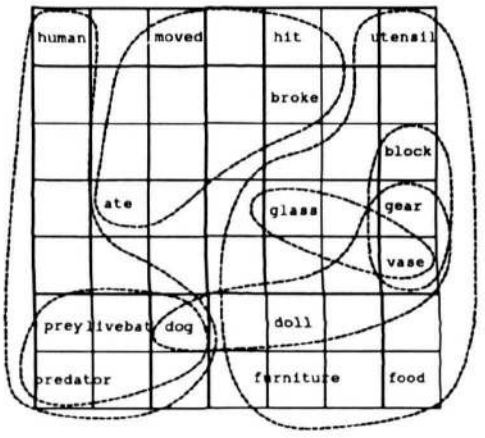


Figure 6: **The semantic map.** The labels on this 7×7 map indicate the maximally responding unit for each semantic word representation. The map is organized according to the semantic categories (table 1).

5.2 Self-organization

The organization of the map, i.e. the assignment of the weight vectors, is formed in an unsupervised learning process [13]. Input items are randomly drawn from the input distribution and presented to the network one at a time (figure 4). The weight vector of the image unit and each unit in its neighborhood are changed towards the input vector, so that these units will produce an even stronger response to the same input in the future. The parallelism of neighboring vectors is increased at each presentation, a process which results in a global order.

The process starts with very large neighborhoods, i.e. weight vectors are changed in large areas. This results in a gross ordering of the map. The size of the neighborhood decreases with time, allowing the map to make finer and finer distinctions between items.

There are several alternatives for implementing the similarity metric, neighborhood selection, and weight change. A biologically plausible process would be based on scalar products of the weight and input vectors, lateral inhibition and redistribution of synaptic resources [14; 20]. These mechanisms can be abstracted and replaced with computationally more efficient ones without obscuring the process itself. The similarity in DISLEX-1 is measured by Euclidian distance, the neighborhood consists of the area around the maximally responding unit, and the weight changes are proportional to the Euclidian difference. More specifically, the output η_{ij} of unit (i,j) is

$$\eta_{ij} = \begin{cases} 1.0 - \frac{\|x - \mu_{ij}\|}{\|x - \mu_{max}\|} & \text{if } (i,j) \in N_c(t) \\ 0.0 & \text{otherwise} \end{cases} \quad (1)$$

where μ_{ij} is the unit's weight vector, x is the input vector, $N_c(t)$ is the neighborhood around the maximally responding unit (shrinking with time), and μ_{max} is the weight vector least similar to x in the neighborhood. This forms a nice concentrated activity pattern around the maximally responding unit. With $\alpha(t)$ as the gain, the weight components are changed according to the input vector – weight vector difference:

$$\Delta\mu_{ij,k} = \begin{cases} \alpha(t)[x_k - \mu_{ij,k}] & \text{if } (i,j) \in N_c(t) \\ 0.0 & \text{otherwise} \end{cases} \quad (2)$$

5.3 Physical and semantic maps

The physical and semantic maps are organized independently, albeit simultaneously, so that associative connections between them can be developed at the same time (see next section). The ordered maps in DISLEX-1 (figures 5 and 6) were obtained in 150 epochs, i.e. by presenting each physical/semantic representation pair (figure 3) to the appropriate map 150 times in random order.

In the self-organizing process, the physical and semantic representations become stored in the weights of the units. For each e.g. physical word, there is an image unit in the physical map, and this unit's weight vector equals the physical representation of that word. The weight vectors of the intermediate units represent combinations of representations. For example, an unlabeled semantic unit between **dog** and **predator** would have features of both domestic and carnivorous animals.

Both maps exhibit hierarchical knowledge organization. Large areas are allocated to different categories of words, and each area is divided into subareas with finer distinctions. The physical map is mainly organized according to the word length. There are separate, adjacent areas for words with 3, 4, 5, 6 and 7 characters. Within these areas, similar words are mapped near each other. For example, **BAT** is mapped between **BOY** and **HIT**, **DOLL** is mapped next to **BALL** etc.

The semantic map has three main areas: verbs, animate objects and inanimate objects. Finer distinctions reveal the semantic categories of table 1. For example, there are subareas for hitters, possessions and fragile-objects, with **vase**, which belongs to all these categories, in the center. Note that the categorization was not directly accessible to the system at any point. It was only manifest in the sentences that were input to the FGREP-mechanism. The

categories were extracted by FGREP, coded into the representations, and finally made visible in the semantic feature map. The final map reflects both the syntactic and semantic properties of the words.

In the self-organizing process, the distribution of the weight vectors becomes an approximation of the input vector distribution [13]. This means that the most frequent areas of the input space are represented to greater detail, i.e. more units are allocated to represent these inputs. For example, the representations for the different animals are very similar (figure 3), yet they accommodate a large area in the map.

The two dimensions of the map do not necessarily stand for any recognizable features of the input space. The dimensions develop automatically to facilitate the best discrimination between the input items. As a result, the ordered areas on the map are likely to have complicated and intertwined, rather than linear shapes.

Feature maps have several useful properties for representing lexical information. (1) The classification performed by a feature map is based on a large number of parameters (the weight components), making it very robust. Incomplete or somewhat erroneous word representations can be correctly recognized. (2) The map is continuous, and can represent items between established categories. In other words, words can have soft boundaries. (3) The differences of the most frequent input items are magnified in the mapping, i.e. the variations of the most common word meanings or surface forms are more finely discriminated. Finally, (4) the self-organizing process requires no supervision and makes no assumptions on the form or content of the words. The properties of the representations which provide the best discrimination are determined automatically.

6 Word associations

6.1 The physical ⇌ semantic mappings

The physical words do not correspond one-to-one to semantic words. Some words have multiple meanings (homonyms), and sometimes the same meaning can be expressed with several different symbols (synonyms). The mapping between the physical and semantic representations is many-to-many.

The training data for DISLEX-1 contained several such ambiguities (figure 3). The physical word **CHICKEN** could mean a living chicken or food. Similarly, **BAT** could be a baseball bat or a living bat. There were also several groups of synonymous words in the data. **MAN, WOMAN, BOY, GIRL** all have the same meaning **human**, **predator** could be **WOLF** or **LION** etc. In the DISLEX model, the many-to-many mapping between the physical words and their meanings is implemented with associative connections between the physical and semantic maps.

6.2 Associative connections

The physical word maps are fully connected to the semantic map with one-directional associative connections (figure 2). There is a connection from each unit in the physical input map to each unit in the semantic map, and from each unit in the semantic map to each unit in the physical output map. The connection weight indicates the strength of the association. The weights are stored as associative output weight vectors per each unit.

The physical and semantic feature maps and the associative connections between them are organized at the same time. The physical pattern for the word is presented to the physical map, and ordinary feature map adaptation takes place. At the same time, the semantic pattern for the same word is input to the semantic map, and the feature map weight vectors in this map are adapted. At this point, both maps display concentrated patterns of activity. DISLEX learns to associate the physical word with its meaning through Hebbian learning. The weights between active units are increased proportional to their activity:

$$\Delta a_{ij,kl} = \alpha(t)\eta_{ij}\eta_{kl} \tag{3}$$

where $a_{ij,kl}$ is the weight between the physical unit (i,j) and the semantic unit (k,l), and η_{ij} and η_{kl} indicate the activities of these units. The associative weight vectors are then normalized, which in effect decreases the weights on all nonactive output connections of the same unit. This corresponds to redistribution of synaptic resources, where the synaptic efficacy is proportional to the square root of the resource [20]. Initially the activity patterns are large, and associative weights are changed in large areas. As the two maps become ordered, the associations become more focused.

For example, DISLEX-1 was trained by simultaneously presenting pairs of physical words and their semantic counterparts from figure 3. The final associative connections form a continuous many-to-many mapping between the two maps. Unambiguous words have focused connections (figures 7a and 8b). If a physical word has several meanings, or one meaning can be expressed with several synonyms, there are several groups of strong connections (figures 7b and 8a). Units located between image units tend to combine the connectivity patterns of nearby words (figure 8a).

7 DISLEX in action

7.1 Transforming representations

A physical word is transformed to its semantic counterpart (and vice versa) through the associative connections. For example in figure 2, the physical representation of **DOG** is input to the physical map, which forms a concentrated activity pattern around the unit labeled **DOG**. The activity propagates through the associative connections (figure 7a) to the semantic map, where a localized activity pattern forms around the unit labeled **dog**. The semantic representation for dog is now output through the weight vector of this unit. In a similar fashion, a semantic representa-

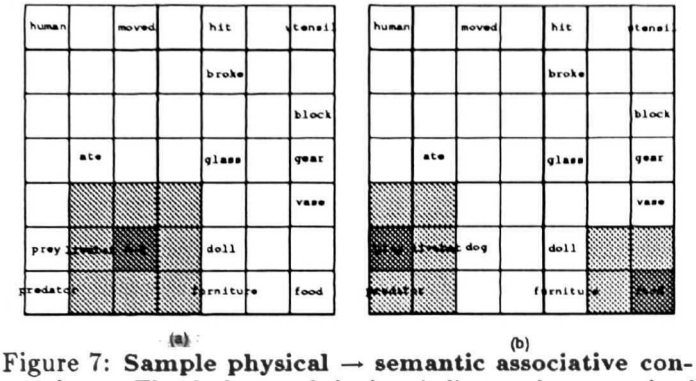


Figure 7: **Sample physical → semantic associative connections.** The darkness of the box indicates the strength of the connection from the physical unit DOG (a) or CHICKEN (b) to the semantic unit. The strongest connections concentrate around the semantic image units. CHICKEN has two possible interpretations, food and prey.

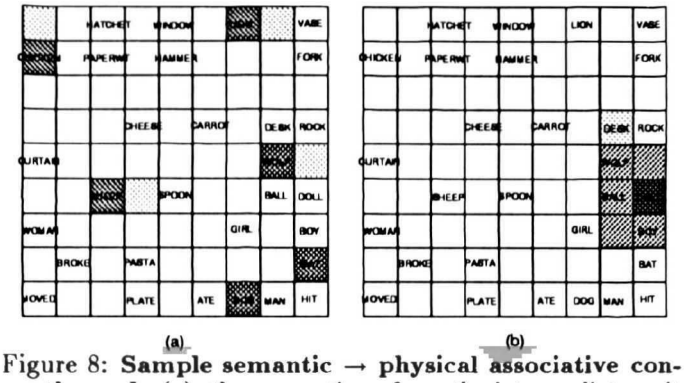


Figure 8: **Sample semantic → physical associative connections.** In (a), the connections from the intermediate unit between dog, livebat, predator and prey are shown. Possible output symbols include all animal names CHICKEN, SHEEP, WOLF, LION, BAT and DOG. In (b), weak connections from doll to nearby units might cause BALL to be output instead of DOLL in noisy conditions.

tion can be transformed to its physical counterpart. The associative connections are different in the two directions, but the same feature map weight vectors are used for both input and output.

The behaviour of the system is very robust. Even if the input pattern is noisy or incomplete, it is usually mapped on the correct unit. Even if this does not happen, the associative connections of the intermediate units provide a mapping that is close enough, so that the correct meaning or symbol can be retrieved with top-down priming.

7.2 Priming

When an ambiguous physical or semantic representation is input to the lexicon, all possible meanings (or symbols) are activated at the same time (figures 7b and 8a). A top-down priming mechanism is employed to select the correct representation. In addition to the associative activity, the map receives priming activation through its input connections. The activities add up, selecting one of the possible interpretations. If the priming arrives after a short delay, all alternatives are briefly active before one of them is selected. This complies with experimental results [24], which indicate that all meanings of ambiguous words are activated upon reading the word.

The expectations generated by the FGREP mechanism provide a possible source for semantic priming. After reading **The wolf ate the**, the FGREP network generates a strong expectation for **prey** [22]. When the physical symbol CHICKEN is read in, both the **food** and **prey** units are initially equally active in the semantic map (figure 7b). The expectation pattern, which is close to the representation for **prey**, is input to the semantic map and summed up with the activity propagated through the associative connections. As a result, the **prey** unit receives the strongest activity and becomes selected.

The weights on the associative connections represent statistical likelihoods of the associations. A very frequently active connection is much stronger than a rare connection. For example, if most of the occurrences of CHICKEN in training DISLEX-1 would have been paired up with **prey**, the CHICKEN unit would tend to activate the **prey** unit much more than the **food** unit. By default, the **prey** meaning would be selected, and stronger priming for **food** would be required to override it.

DISLEX-1 simply selects and outputs the representation stored at the maximally responding unit. The selection could also be implemented with lateral inhibition, where the map settles into a localized response around the maximally responding unit [20]. The settling times would most likely correspond to the reaction times observed in humans [23]. High-frequency words would have shorter reaction times, and these times could be changed with priming. With several equally likely interpretations, settling would take longer.

7.3 Errors

The DISLEX architecture is well suited into modeling dyslexic performance errors. If the system performance is degraded e.g. by adding noise to the connections, two types of input errors and two types of production errors are observed.

In the input, a physical representation may be mapped incorrectly on a nearby unit in the physical map. This corresponds to reading or hearing the word incorrectly. For example, DOLL may be input as BALL (figure 5). The activity in the physical map may also propagate incorrectly to a nearby unit in the semantic map, in which case e.g. CHICKEN would be understood semantically as livebat (figure 7b).

Analogously in production, a semantic input representation can be classified incorrectly, and a word with a similar but incorrect meaning is produced. For example, if the semantic pattern for **block** is accidentally mapped on **vase** (figure 6), the output reads VASE instead of, say, PAPERWT. Or, the activity in the semantic map may be propagated incorrectly to the physical map, and a word with a similar surface form but different meaning is output. This means generating BALL instead of DOLL (figure 8b).

Errors of this kind occur in noisy, stressful or overload situations in normal human performance. They are also documented in patients with deep dyslexia [4; 3]. The observed visual and semantic paralexic errors can be explained by above mechanisms, giving strong support to the physical/semantic feature map architecture.

If priming is used in the model, there is also a possibility for another type of error, the Freudian slip. This occurs when very strong semantic priming interferes with the output function. For example, if `doll` is input to the semantic map, together with simultaneous priming for **gear**, the activity is propagated through the associative connections of both. As a result, the physical `BALL` might receive the strongest activation, and would be output instead of `DOLL`. The output symbols are similar, but the meaning of `BALL` reveals the semantic priming.

8 Modeling aphasia

The DISLEX architecture is in good agreement with the current theories of the human lexical system [3; 27; 26]. Many observed lexical deficits in acquired aphasia have straightforward explanations in the model.

A common feature of the aphasic deficits is category specificity. The patient may have difficulties only with words belonging to a specific syntactic or semantic category. In certain patients the lexical access to e.g. function words is selectively impaired, in other cases the patient has trouble with verbs [3; 4]. More specific impairments seem to occur in semantic hierarchies. Some patients have trouble with e.g. concrete words, or inanimate objects [28], or even as specific classes as names of fruits and vegetables [10].

Deficits of this kind can be explained by the topological organization of the semantic memory. The semantic map in DISLEX is hierarchically organized, and reflects both the syntactic and semantic properties of the words. Localized lesions to the map produce selective impairments, like the above.

In some cases the impairments cover all modalities, sometimes they are limited only to verbal input or output, or even only to orthographic or phonological domain. This suggests that the semantic memory, visual input, and verbal input/output modalities are represented in separate structures, strongly supporting the distributed DISLEX architecture.

For example, some patients were unable to access the specific meanings from verbally as well as visually (with pictures) presented cues [26; 28]. This implies that the semantic memory itself, i.e. the map, had been damaged. Another patient could not give definitions for aurally presented names of living things such as "dolphin", although he was able to describe other objects. But when shown a picture of a dolphin, he could name it and give an accurate verbal description of it [16]. This suggests that the visual pathway to the semantic memory, the semantic memory itself, and the verbal output were preserved, but the verbal access to the semantic memory had been damaged. In another case, the patient was unable to name fruits and vegetables, although he was able to match their names with pictures, and classify them correctly when their names were presented aurally [10]. In other words, his semantic memory and verbal input were preserved, and the verbal output function was selectively impaired.

The impairment of semantic categories which is restricted to a single input or output modality can be explained in DISLEX by severed pathways between physical and semantic maps. The pathways are not single axons, but consist of interneurons, which also exhibit map-like organization. Close to the semantic map, the organization is semantic, close to the physical map it parallels the physical map. If the pathway is severed close to the semantic map, semantic impairment within this modality results.

The dissociation of the orthographic and phonological modalities is also well-documented. Some patients have deficits only in one of the input or output channels, or different deficits in different channels [2]. For example, a patient may have spelling difficulties exclusively in the orthographic output domain [9; 18]. The types of errors in visual and phonological dyslexia (section 7.3) further indicate that the channels are organized according to the physical forms of the words. The DISLEX model predicts that it would be possible to lose access to specific types of physical symbols, as a result of localized damage to a physical map.

In the aphasic impairments, the high-frequency words are often better preserved than rare words. This is also predicted by the feature map organization. The most often occurring words occupy larger areas in the map, making them more robust against damage.

9 Discussion

The DISLEX model can be locally lesioned, and it displays deficits similar to human patients. This suggests that the model successfully represents some of the physical structure underlying the lexical system in the brain. The architecture is based on word maps, where different units are selectively sensitive to different words in the data. Several low-level sensory maps are known to exist in the central nervous system, e.g. retinotopic maps, tonotopic maps, and also tactile and motor maps. Recently it was found that neurons in the hippocampus respond selectively to visually presented words [11]. These response characteristic could be explained by a map-like structure.

DISLEX still finesses much of the fine neural structure, and the mapping to the neuron level is nontrivial. The units and connections in the model do not necessarily correspond one-to-one to neurons and synapses, but rather, to connected groups of neurons. For example, the weight vectors in the maps are used both for input and output, which is not a plausible model of the synaptic efficacies. However, these two-way connections could be implemented with tightly interconnected (or phase-locking) groups of neurons in the brain.

The associative connections between two feature maps

learn a many-to-many mapping from one distributed representation space to another, which is hard to do with other neural network mechanisms such as backpropagation. In the maps, several representations can be active at the same time, whereas e.g. in an assembly-based representation all the different alternatives would be combined into a single average representation pattern [22].

DISLEX is primarily a model of single word processing. It does not have special mechanisms for representing and processing phrasal structures and morphology. There are two possible ways of doing this, and it seems that both of them are involved. Common morphological forms and phrases, such as `nationalism` or `The Big Apple` could be represented like words, as single entries in the physical and semantic maps. More complex phrases and unusual, constructive forms, e.g. `kick the bucket` or `non-preemptive` could be represented in the lexicon by their constituents, and parsed/generated by a higher-level language processing module.

10 Conclusion

The DISLEX architecture models the human lexical system at the level of physical structures. The architecture accounts for many observed dyslexic performance errors and lexical deficits in acquired aphasia. DISLEX-1, the AI implementation of the model, can be used as an input/output filter for a natural language processing system, which communicates with the external world with physical symbol representations, but internally processes semantic representations.

References

[1] Yigal Arens. *CLUSTER: An Approach to Contextual Language Understanding.* PhD thesis, Computer Science Division, University of California, Berkeley, 1986.

[2] A. Basso, A. Taborelli, and L. A. Vignolo. Dissociated disorders of speaking and writing in aphasia. *Journal of Neurology, Neurosurgery and Psychiatry*, 41:526–556, 1978.

[3] Alfonso Caramazza. Some aspects of language processing revealed through the analysis of acquired aphasia: The lexical system. *Annual Reviews in Neuroscience*, 11:395–421, 1988.

[4] Max Coltheart, Karalyn Patterson, and John C. Marshall, editors. *Deep Dyslexia. International Library of Psychology*, Routledge and Kegan Paul, 1980.

[5] Garrison W. Cottrell and Steven L. Small. A connectionist scheme for modelling word sense disambiguation. *Cognition and Brain Theory*, 6(1):89–120, 1983.

[6] Michael G. Dyer. *In-Depth Understanding: A Computer Model of Integrated Processing for Narrative Comprehension.* MIT Press, Cambridge, MA, 1983.

[7] Michael Gasser. *A Connectionist Model of Sentence Generation in a First and Second Language.* PhD thesis, Computer Science Department, UCLA, 1988.

[8] Helen Gigley. Process synchronization, lexical ambiguity resolution and aphasia. In Steven L. Small, Garrison W. Cottrell, and Michael K. Tanenhaus, editors, *Lexical Ambiguity Resolution*, Morgan Kaufmann Publishers, Los Altos, CA, 1988.

[9] R. A. Goodman and Alfonso Caramazza. Aspects of the spelling process: Evidence from a case of acquired dysgraphia. *Language and Cognitive Processes*, 1(4):263–296, 1986.

[10] John Hart, Rita Sloan Berndt, and Alfonso Caramazza. Category-specific naming deficit following cerebral infarction. *Nature*, 316(1):439–440, August 1985.

[11] Gary Heit, Michael E. Smith, and Eric Halgren. Neural encoding of individual words and faces by the human hippocampus and amygdala. *Nature*, (333):773–775, 1989.

[12] Alan H. Kawamoto. Distributed representations of ambiguous words and their resolution in a connectionist network. In Steven L. Small, Garrison W. Cottrell, and Michael K. Tanenhaus, editors, *Lexical Ambiguity Resolution*, Morgan Kaufmann Publishers, 1988.

[13] Teuvo Kohonen. *Self-Organization and Associative Memory*, chapter 5. Springer-Verlag, Berlin; New York, 1984.

[14] Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, (43):59–69, 1982.

[15] Geunbae Lee, Margot Flowers, and Michael G. Dyer. Learning distributed representations of conceptual knowledge and their application to script-based story processing. *Connection Science*, 1990. (In press).

[16] Rosaleen A. McCarthy and Elizabeth K. Warrington. Evidence for modality-specific meaning systems in the brain. *Nature*, 334(4):428–430, August 1988.

[17] James L. McClelland and Alan H. Kawamoto. Mechanisms of sentence processing: Assigning roles to constituents. In James L. McClelland and David E. Rumelhart, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume II: Psychological and Biological Models*, MIT Press, 1986.

[18] G. Miceli, M. C. Silveri, and Alfonso Caramazza. Cognitive analysis of a case of pure dysgraphia. *Brain and Language*, 25:187–212, 1985.

[19] Risto Miikkulainen. *A Neural Network Model of Script Processing and Memory.* Technical Report UCLA-AI-90-03, Artificial Intelligence Laboratory, Computer Science Department, University of California, Los Angeles, 1990.

[20] Risto Miikkulainen. *Self-Organizing Process Based on Lateral Inhibition and Weight Redistribution.* Technical Report UCLA-AI-87-16, Artificial Intelligence Laboratory, Computer Science Department, UCLA, 1987.

[21] Risto Miikkulainen and Michael G. Dyer. Encoding input/output representations in connectionist cognitive systems. In David S. Touretzky, Geoffrey E. Hinton, and Terrence J. Sejnowski, editors, *Proceedings of the 1988 Connectionist Models Summer School*, Morgan Kaufmann Publishers, 1989.

[22] Risto Miikkulainen and Michael G. Dyer. Natural language processing with modular neural networks and distributed lexicon. 1989. Submitted to *Cognitive Science.*

[23] Greg B. Simpson and Curt Burgess. Activation and selection processes in the recognition of ambiguous words. *Journal of Experimental Psychology: Human Perception and Performance*, 11(1):28–39, 1985.

[24] D. A. Swinney. Lexical access during sentence comprehension: (Re)consideration of context effects. *Journal of Verbal Learning and Verbal Behavior*, 18:645–659, 1979.

[25] David L. Waltz and Jordan B. Pollack. Massively parallel parsing: A strongly interactive model of natural language interpretation. *Cognitive Science*, (9):51–74, 1985.

[26] Elizabeth K. Warrington. The selective impairment of semantic memory. *Quarterly Journal of Experimental Psychology*, 27:635–657, 1975.

[27] Elizabeth K. Warrington and Rosaleen A. McCarthy. Categories of knowledge: Further fractionations and an attempted integration. *Brain*, 110:1273–1296, 1987.

[28] Elizabeth K. Warrington and T. Shallice. Category specific semantic impairments. *Brain*, 107:829–854, 1984.

[29] Uri Zernik. *Strategies of Language Acquisition: Learning Phrases from Examples in Context.* PhD thesis, Computer Science Department, University of California, Los Angeles, 1987. Technical Report UCLA-AI-87-1.

Efficient Learning of Language Categories: The Closed-Category Relevance Property and Auxiliary Verbs

Sheldon Nicholl and David C. Wilkins

nicholl@cs.uiuc.edu
wilkins@cs.uiuc.edu
Department of Computer Science
University of Illinois
405 North Mathews Avenue
Urbana, IL 61801

Abstract

This paper describes the mechanism used by the ALACK language acquisition program for identification of auxiliary verbs. Pinker's approach to this problem (Pinker, 1984) is a general learning algorithm that can learn any Boolean function but takes time exponential in the number of feature dimensions. In this paper, we describe an approach that improves upon Pinker's method by introducting the Closed-Category Relevance Property, and showing how it provides the basis of an algorithm that learns the class of Boolean functions that is believed sufficient for natural language, and does not require more than linear time as feature dimensions are added.

1 Introduction

Within the study of language acquisition, the problem of category identification is still a challenge to formal theories of language acquisition. Even the identification of the members of a closed category such as the Auxiliary Verbs (hereinafter referred to as AUX) stands unresolved. The principal approaches to this problem include (Anderson, 1983), (Berwick, 1985), and (Pinker, 1984); Our approach is closest to that of Pinker. We share with Pinker the following five assumptions: (1) Language is learned not from a string of words alone but from the corresponding meaning (and possibly other attributes) as well. (2) Some components of the meaning can be represented with features. (3) The features are drawn from sets we will call *feature dimensions*. Examples of several feature dimensions are shown below in Table 1. (4) Candidate AUXes are not annotated with syntactic features in the input, e.g. (Berwick, 1985), nor is the input prechunked into phrase-like groupings, e.g., (Anderson, 1983). (5) Steele's cross-linguistic generalization (Steele et al., 1981), holds for AUXes: AUXes encode tense or modality or both.

Pinker's approach is based on a general learning algorithm that (i) can learn any Boolean function but (ii) takes time exponential in the number of feature dimensions. In contrast, our approach, which depends on the Closed-Category Relevance Property, defined Section 2, (i) cannot learn an arbitrary Boolean function but is conjectured to be sufficient for natural language, and (ii) does not require more than linear time as feature dimensions are added.

2 The Closed-Category Relevance Property

In this section, we introduce the Closed-Category Relevance Property that will be used to identify the category AUX; this is the basis of our algorithm for AUX identification. Our goal is identification only, not full control. So to recognize that **are**, for example, is an auxiliary is sufficient; to learn the rules that control when to choose **are** instead of **is** or **were**, for example, is beyond the scope of this paper.

The Closed-Category Relevance Property can be viewed abstractly as a two-place predicate R which takes a word as its first argument and a feature dimension as its second argument: $R(word, dimension)$. Its purpose is to relate words to features that control their usage. It is meant to capture the idea that *animacy*, for example, has no bearing on the word **are**, i.e., $\neg R(\textbf{are}, animacy)$, but that *tense*, for instance, does: $R(\textbf{are}, tense)$.

The Closed-Category Relevance Property is defined as follows: $R(word, dimension)$ is true **iff** *word* can encode some but not all of the values in *dimension*. $R(\textbf{are}, tense)$ is true because **are** can encode present and future time but not past time. The definition is satisfied because **are** can encode some but not all of the values on the tense dimension. $R(\textbf{are}, animacy)$ is false bacause **are** can be used in both animate ("The dogs are running") and inanimate ("The computers are running") contexts, so it trivially encodes both the values on the animacy dimension.

3 The AUX Learning Algorithm

The Closed-Category Relevance Property has been embodied in a computer model of language acquisition called ALACK, which runs in Common Lisp on a Sun 4.

3.1 The Input to the Algorithm

The input to ALACK relevant to this discussion consists of: (1) a segmented string of words, each of which is segmented into grammatical morphemes, (2) a list of semantic categories corresponding to each word, drawn from the set {`thing`,`event`,`state`,`null`} where `thing` marks a perceptually salient physical object, `event` marks a perceptually salient event, action, or process, `state` marks some ongoing state, and `null` is the default which applies to all other words, and (3) a list of sets of feature values, which are empty sets for words marked `null`, nonempty otherwise.

3.2 The Algorithm and an Example

In this section, we describe the steps of our Closed-Category Relevance algorithm, and show how each step processes the following input that is based on actual input to ALACK:

person	*1st-person, 2nd-person, 3rd-person*
number	*singular, plural*
tense	*past, present, future*
animacy	*animate, inanimate*
modality	*modal, nonmodal*
aspect	*perfective, imperfective, progressive*

Table 1: Features used as examples throughout the text. They, and the actual values within them, are used as illustration in the text, and of course do not constitute a claim as to what is indeed linguistically complete and correct.

```
(((the) (men) (are) (walk ing))
 (null thing null event)
 (() (3rd-person plural animate) () (present progressive)))
```

The feature values are drawn from the same sets of *feature dimensions* described in the introduction. ALACK embodies the Closed-Category Relevance Property in a manner logically equivalent to the following.

- Step 1. All words marked null are collected into a set E. In this case, $E = \{\textbf{are}, \textbf{the}\}$. It is this set that is tested for closed-category relevance; the treatment of the other words is beyond the scope of this paper.

- Step 2. All the feature values are collected into a set F. In this case, $F = \{$*3rd-person, plural, animate, present, progressive*$\}$. ALACK filters the input to make sure that F contains only one value from each feature dimension. Inputs violating this constraint are ignored.

- Step 3. Unless they have been built already, ALACK constructs all the triples $E \times G \times H$ where

$$G = \{x \mid x \subseteq F \wedge |x| = 1\}$$

$$H = \{dim(f) \mid f \in F\}$$

$$dim(f) = y \quad \text{if } f \in y$$

  For example, $dim($*singular*$) =$ *number*, since *number* $= \{$*singular, plural*$\}$. Three examples of $E \times G \times H$ are $($*are*$, \{$*present*$\},$ *tense*$)$, $($*are*$, \{$*animate*$\},$ *animacy*$)$, and $($*the*$, \{$*plural*$\},$ *number*$)$.

- Step 4. If a triple (e, S, d) where $e \in E$ and $d \in dim(F)$ has already been constructed, set S is updated to include the new feature value $f \in F$: $S_{new} = S_{old} \cup \{f\}$. So for example, if ALACK gets the following input

```
(((the) (ball) (is) (fall ing))
 (null thing null event)
 (() (3rd-person singular inanimate) () (present progressive)))
```

  then the example triple $($*are, $\{$animate$\}$, animacy*$)$ is updated to

$$(are, \{animate, inanimate\}, animacy)$$

- Step 5. All the updated triples are tested against the following rule:

$$(x, y, y) \Rightarrow \neg R(x, y)$$

  Since $animacy = \{animate, inanimate\}$, the example triple matches the left-hand side, which forces the conclusion $\neg R($**are**$, animacy)$; i.e., **are** is not closed-category relevant for animacy, as discussed above. But **are** is still relevant for *tense*: the triple $(are, \{present\}, tense)$ has not been changed. Pinker's model and ours agree that all dimensions not explicitly found to be irrelevant are relevant by default.

ALACK organizes some feature dimensions into a set, or *domain*, which we will call the *verbal* domain. In the current implementation of ALACK, this domain is as follows:

$$verbal = \{tense,\ aspect\}$$

Formally, category inference is done as follows. Given morpheme x and dimension y,

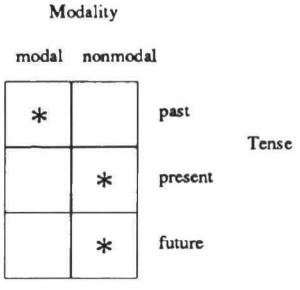
$$R(x, y) \wedge (y \in verbal) \Rightarrow x \in AUX$$

The rule is justified by the previous discussion on Steele's generalization (Steele et al., 1981). ALACK's implementation does not yet include modality; aspect has been included since it works well for English. For example, since $R(\textbf{are}, tense)$ and $tense \in verbal$, the rule allows **are** $\in AUX$ to be concluded.

4 Analysis of AUX Learning Algorithm

Correctness. Clearly the closed-category relevance property leads to an algorithm that is **not** correct for all Boolean functions. For example, suppose that English had an auxiliary **hawn** whose paradigm, or grammar chart, looked like this:

"HAWN":



where the stars mean to use the auxiliary, and the blanks mean ϕ (the null morpheme), for instance. The auxiliary **hawn** is irrelevant along both the *tense* and *modality* dimensions, even though both dimensions are (1) important to auxiliary identification, as discussed below, and (2) important to the proper usage of **hawn**. Though Closed-Category Relevance is not correct from the standpoint of full logical generality, it is only a small stipulation beyond the generalization of (Steele 1981); English has nothing like **hawn**. This leads to the **Closed-Category Relevance Conjecture:** All the auxiliaries in all the world's languages that have auxiliaries are relevant for tense or modality or both. This Conjecture goes beyond Steele's generalization in that the Conjecture would not allow something like **hawn**, while Steele's generalization would. Steele's generalization is satisfied by any encoding of tense or modality; the Conjecture demands that the encoding satisfy the Relevance Property as well.

Complexity Analysis. We wish to examine the time complexity of determining $\neg R(w, d)$ for a given w, d as the total number of dimensions T in *features* is increased. For the sake of this argument, we can stipulate that in the implementation, the triples (e, S, d) are accessed by their dimension via a function h, where $h(d) = \{(e, S, d') \mid d' = d\}$. If h is chosen to be implemented in an array A, one may simply search A linearly for the desired dimension d, i.e., the search time is $O(T)$. If A is sorted, the time drops to $O(\log(T))$. If h is a hash function, that time can be lowered significantly.

5 Cross-Linguistic Analysis of Relevance

The idea of Closed-Category Relevance has interesting implications when applied to Bickerton's work (Bickerton, 1984). Suppose we are given the problem of identifying the auxiliaries in some

of the creole languages that Bickerton has studied. For instance, in Hawaiian Creole, three auxiliaries to try are **bin**, **go**, and **stei**. For Lesser Antilles Creole, the auxiliaries are **ka**, **ke**, and **te**. Two of the auxiliaries in Saramaccan are **ta**, and **bi-o-ta**. It is possible to translate his notation for tense, modality, and aspect (see Table 1, p. 183) into this system in the following way. Let $features = verbal = \{tense, modality, aspect\}$ where $tense = \{anterior, nonanterior\}$, $modality = \{realis, irrealis\}$, and $aspect = \{punctual, nonpunctual\}$. Then for Hawaiian creole, *R(***stei***, aspect)*, *R(***go***, modality)*, *R(***bin***,tense)*, and for Lesser Antilles Creole, *R(***ka***,aspect)*, *R(***ke***,modality)*, and *R(***te***,tense)*. Now in a case like Saramaccan where auxiliaries are built up morphologically, closed-category relevance can be applied successfully to each morpheme individually or to a whole word, e.g. *R(***bi-o-ta***,tense)* or *R(***ta***, aspect)*. Closed-Category Relevance is confirmed for auxiliaries in these languages. Although the above demonstration hardly constitutes a full confirmation of Closed-Category Relevance, it does show that further tests of Relevance in other languages are worthwhile. It also lends credence to the idea of Closed-Category Relevance as an acquisition principle.

6 Comparison to Pinker's Learning Method

This section will do three things: show that Pinker's Method is logically correct, analyze its computational complexity, and compare it to Closed-Category Relevance.

6.1 Correctness

The set of features described in Table 1 can be viewed as an instance of the following format, where the given set of features is simply a set of dimensions, and each dimension is simply a set of values. To make the statement of Pinker's method more precise, we introduce here a set of *feature-names*, with the obvious bijection between *features* and *feature-names*: $features = \{dim_1, dim_2, \ldots, dim_n\}$, $dim_i = \{val_1, val_2, \ldots, val_{f(i)}\}$, and $feature\text{-}names = \{d_1, d_2, \ldots, d_n\}$. The input to Pinker's procedure is first a sequence of *attribute-lists*, which we will index with $q \in \{1, 2, 3, \ldots\}$. Each attribute-list is a set of pairs, with a feature name in the first slot of the pair and a value from the corresponding feature dimension in the second:

$$attribute\text{-}list_q = \{(d_{q_1}, val_{q_1}), (d_{q_2}, val_{q_2}), \ldots, (d_{q_n}, val_{q_n})\}$$

The procedure is also provided with a morpheme m_i on each trial, where

$$m_{i_q} \in M = \{m_1, m_2, \ldots, m_i, \ldots, m_{g(q)}\}$$

The goal is to discover (learn) the Boolean expression B_j for each m_i, where B_j is a possibly complex Boolean expression built up from the pairs in the *attribute-lists*, using conjunction and disjunction only, without negation, and $m_i \Leftrightarrow B_j$.

Pinker's solution to this learning problem is to build a big multi-dimensional array, a *Paradigm*, and to fill single array locations by (1) reading each input q as a set of coordinates and (2) placing m_{i_q} at the location specified by these coordinates. The array is built up one dimension at a time; the dimensions to add are selected at random from the image of $\{x \mid (x, y) \in attribute\text{-}list_q\}$, for some q. That is, the array can be built up only from dimensions that occur somewhere in the input. Pinker does not say what to do with the morpheme entries when a new dimension is added; one possibility would be simply to forget them all and start over with a bigger matrix. We will neglect this problem and assume that we begin with a big enough matrix.

Given a sufficiently large matrix, Pinker's method is correct, since it is simply storing the examples into the matrix as they come in. Although this result is new (Pinker didn't give correctness proofs), it is very minor, and serves only to provide a background for the next result, which is the complexity result.

6.2 Complexity

Since Pinker's learning procedure P chooses feature dimensions dim_i at random, it is quite possible that P will choose an irrelevant dimension, say dim_5, even though B only requires the use of some other dimensions, say dim_1, dim_2, and dim_3. It is important to dispose of irrelevant dimensions, since, among other things, Pinker's method for finding auxiliaries does not tolerate irrelevant dimensions. We will first give Pinker's statement of his method for disposing of irrelevant dimensions, show its correctness, and then give its time complexity.

Here is Pinker's (Pinker, 1984) method for eliminating irrelevant dimensions, which he calls Procedure I3. His use of "cells" corresponds to "array locations" in this paper; similarly, "paradigm" means "array", and "affix" means "morpheme":

> If the same affix appears in all the cells defining a dimension across a given combination of values of the other dimensions, and this is true for every possible combination of values of the other dimensions, eliminate that entire dimension from the paradigm. (Pinker, 1984) p. 186.

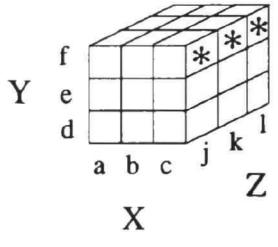
For example, suppose that $features = \{X, Y, Z\}$, $X = \{a, b, c\}$, $Y = \{d, e, f\}$, $Z = \{j, k, l\}$, and $feature\text{-}names = \{\mathbf{X},\mathbf{Y},\mathbf{Z}\}$, and suppose further that the input consists of the following three input attribute-lists for morpheme m:

$$attribute\text{-}list_1 = \{(\mathbf{X}, c), (\mathbf{Y}, f), (\mathbf{Z}, j)\}$$

$$attribute\text{-}list_2 = \{(\mathbf{X}, c), (\mathbf{Y}, f), (\mathbf{Z}, k)\}$$

$$attribute\text{-}list_3 = \{(\mathbf{X}, c), (\mathbf{Y}, f), (\mathbf{Z}, l)\}$$

The (complete) array resulting from these inputs is:



where the asterisks correspond to the three inputs. Pinker's elimination method says that the Z dimension can be eliminated from the array, leaving just dimensions X and Y. We now show that this is logically correct.

Note that the possible attribute-lists for m as shown in the array can be represented as follows:

$$cfj \vee cfk \vee cfl = cf(j \vee k \vee l) = cf(true) = cf$$

since $j \vee k \vee l \Rightarrow true$. Now since the resulting expression, cf, contains no reference to the Z dimension, that dimension can be eliminated. It should now be clear that this argument can be carried out in general for any number of dimensions, values, and inputs.

Procedure I3 can now be stated formally. Let $S = \{f_1, f_2, \ldots, f_k\}$ be a set of feature dimensions. Let $B(S)$ be a Boolean expression based on the members of S. Given m iff $B(S)$, dimension $f_i \in S$ can be eliminated from S only if $[B(S)$ *iff* $B(S - \{f_i\})]$. In our example, $S = \{X, Y, Z\}$ and $f_i = Z$. So

$$B(S) = B(\{X, Y, Z\}) = cfj \vee cfk \vee cfl = cf = B(\{X, Y\}) = B(S - \{f_i\})$$

This is just a formal way of saying that dimension Z can be eliminated from consideration, just as it was in the array realization above.

Result: Procedure I3 is NP-complete. Proof: The elimination of f_i from S requires that the expression $B(S) \Longleftrightarrow B(S - \{f_i\})$ be shown to be a tautology. The tautology problem is NP-complete.

This result will be used shortly. First, a predicate similar in spirit to our Closed-Category Relevance predicate can be defined: $R'(word, dimension)$ iff $(word \Leftrightarrow B(S)) \wedge (dimension \in S)$. Pinker's AUX identification method can now be approximated by the following expression. Pinker left several terms mathematically undefined.

$$Prob(R'(word, tense) \vee R'(word, modality), Phon(word), Syn(word)) \Rightarrow word \in AUX$$

That is, a word is an auxiliary if it satisfies an undefined predicate ($Prob$) based on a probabilistic combination of its arguments: the ability of the word to encode tense or modality (R'), an undefined predicate ($Phon$) based on certain phonological properties of the word, and an undefined predicate (Syn) based on certain other syntactic properties of the word. Now if $Prob$ is strict in its first argument, $Prob$ must be at least NP-complete. Hence Pinker's auxiliary identification procedure is at least NP-complete under the assumption that $Prob$ needs the output of R'.

6.3 Comparison to Closed-Category Relevance

Figure 1 showed that there exist logical, if not linguistic, counterexamples to Closed-Category Relevance. By contrast, section 6.1 showed that under assumptions (1) – (5) in section 1, Pinker's procedure could handle any logically possible AUX rule, including Figure 1. Section 4 showed, however, that Closed-Category Relevance leads to an algorithm that is fast, while section 6.2 showed that Pinker's procedure is NP-complete.

7 Conclusion

The Closed-Category Relevance Property has been defined and has been shown to lead to an efficient algorithm for the identification of Auxiliary Verbs. Relevance was shown to hold in several languages other than English, demonstrating that the same algorithm could be applied to Auxiliary Verb identification in those languages. Finally, Pinker's method was subjected to an analysis which, under certain reasonable assumptions, proved it logically correct but NP-complete.

Acknowledgements

The authors would like to thank Cindy Fisher, Peter Haddawy, and Steve Pinker for their very helpful comments.

References

Anderson, J. R. (1983). *The Architecture of Cognition.* Cambridge, MA: Harvard.

Berwick, R. C. (1985). *The Acquisition of Syntactic Knowledge.* Cambridge, MA: MIT Press.

Bickerton, D. (1984). The language bioprogram hypothesis. *The Behavioral and Brain Sciences*, 7:173–221.

Pinker, S. (1984). *Language Learnability and Language Development.* Cambridge, MA: Harvard.

Steele, S., Akmajian, A., Demers, R., Jelinek, E., Kitagawa, C., Oehrle, R., and Wasow, T. (1981). *An Encyclopedia of AUX: a study of cross-linguistic equivalence.* Cambridge, MA: MIT Press.

The Representation of Word Meaning

Renison J. Gonsalves, Brooklyn College

Abstract: This article shows that a substantial portion of the empirical evidence regarding the representation of word meaning can be explained by the definitional semantic theory of Jerrold J. Katz (henceforth ST). First, we look at the relative complexities of four types of negatives which, according to Fodor, Fodor, and Garrett (1975; henceforth FFG), show that definitions are not psychologically real. The ST definitional structures explain the FFG results in terms of the number of disjuncts generated by the negative elements in their ST representations. Next we look at the arguments of Gentner for componential structure from evidence from a recall experiment that considers connectedness relationships between the noun phrases of sentences with three types of verbs. Gentner's results can be explained in terms of the number of argument places in the ST representations, and the same explanation can be used with respect to evidence from studies by Fodor, Garrett, Walker, and Parkes (1980; henceforth FGWP) and Gergely and Bever (1986).

John Locke devotes a great deal of An Essay Concerning Human Understanding (1689) to his account of the mental representation of word meaning, and this account is essentially definitional or decompositional in character:

> The Mind being, as I have declared, furnished with a great number of simple *Ideas*, conveyed by the *Senses*, as they are found in exterior things, or by *Reflection* on its own Operations, takes notice also, that a certain number of these simple *Ideas* go constantly together; which being-presumed to belong to one thing, and Words being suited to common apprehensions, and made by use for quick dispatch, are called so united by one subject, by one name; . . . (II, xxiii, 1)

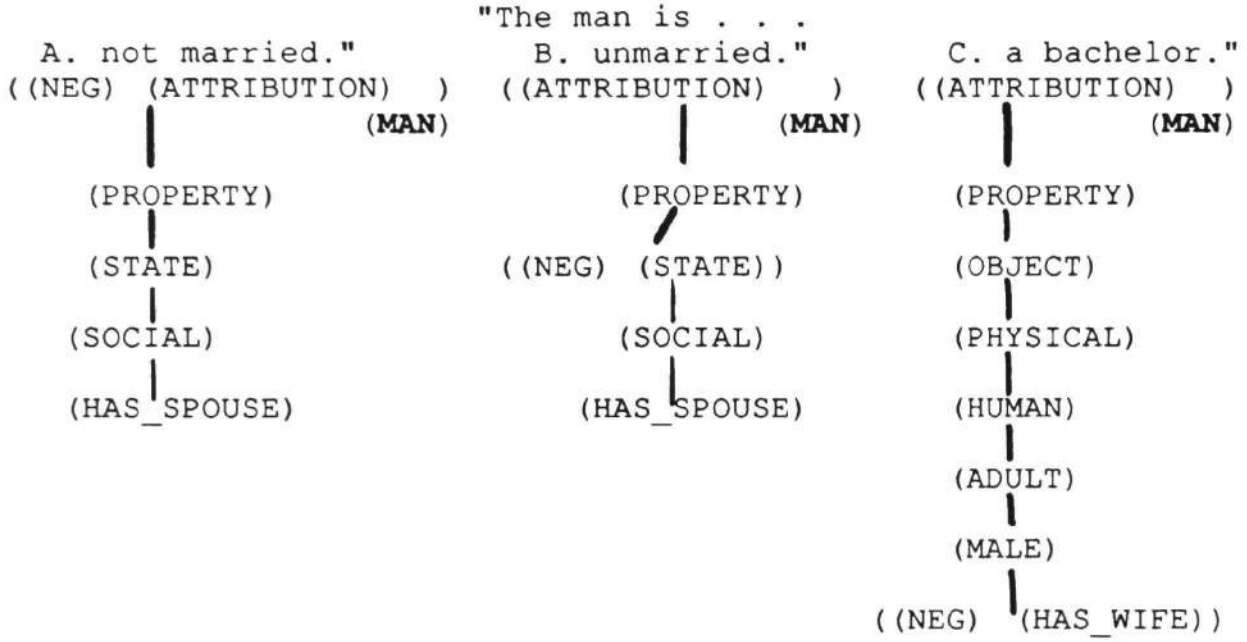
Contemporary versions of this definitional approach came into such substantial criticism (Fodor et al, 1975 and 1980) that for most of the 1980s many investigators in this area worked under the erroneous assumption that it had been completely discredited. However, three hundred years after the first publication of Locke's Essay, in a thorough critical review of all of the evidence and arguments, McNamara and Miller (1989) came to a pro-decompositional view, concluding that "there's little evidence against conceptual structure and there's some evidence for it." In this paper I will show that a substantial portion of the empirical evidence can be perspicuously explained by the definitional semantic theory of Katz (1972, 1976, 1989; henceforth ST).

Part 1: Fodor, Fodor, and Garrett and the Complexity of Negatives

One of the major components of the FFG argument against the psychological reality of definitional structures was the results of a psycholinguistic study involving four types of negatives: explicit negatives, e.g. "not married"; implicit negatives, e.g. "doubt," deny," fail"; morphological negatives, e.g. "unmarried"; and pure definitional negatives (PDNs) e.g. "bachelor." Subjects

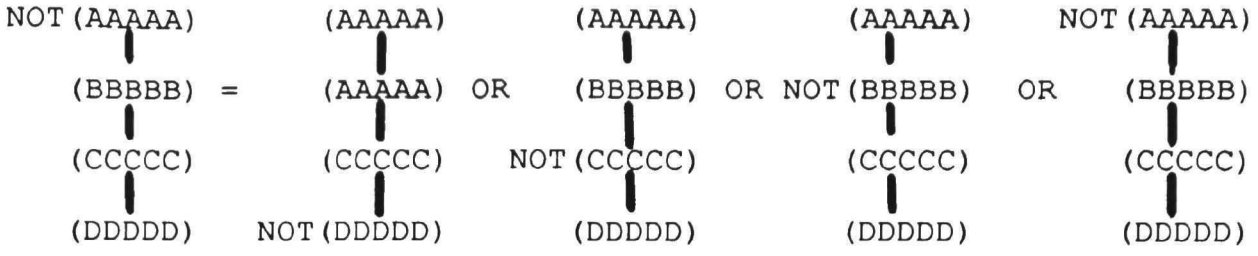
were presented with the sentence "If practically all of the men in the room are _____, then few of the men in the room have wives," with either "not married," "unmarried," or "bachelor" in the blank. The subjects were then asked to judge the correctness of the inference and response times were measured. FFG found that explicit negatives were the most difficult; morphological and implicit negatives were both less difficult; and PDNs were the least difficult. Based on this evidence FFG argued that "PDNs do not act as though they contain a negative element," and that "PDNs are not semantically analyzed," contrary to the definitional view.

However, in ST we might have the following representations:



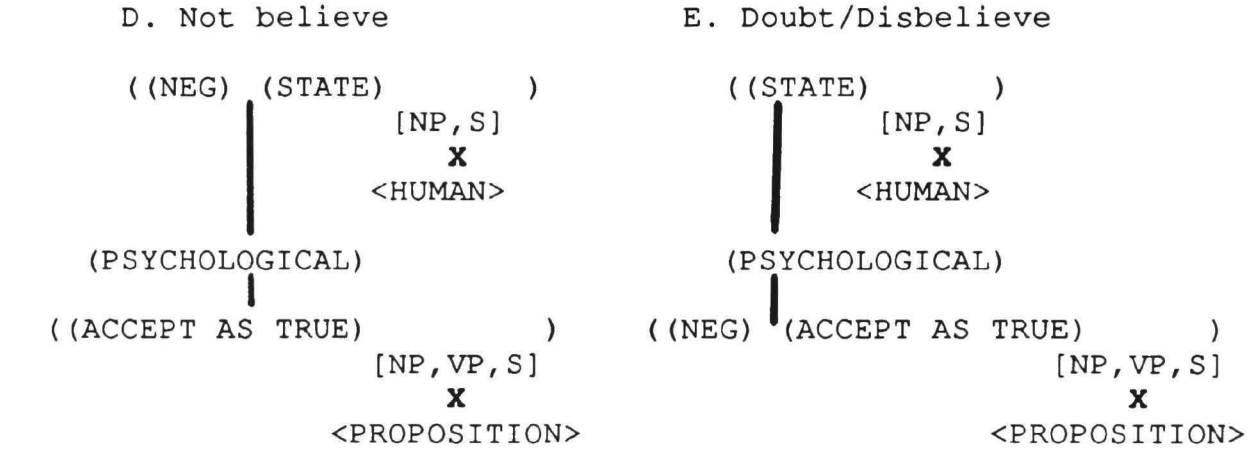
Our reading for sentence A says that the attribution to the man of the property of being in a social state of having a spouse is negated, while our reading for B says that there is an attribution to the man of the property of not being in the social state of having a spouse, and finally our reading for C says that there is an attribution to the man of the property of being a physical, object, that is human, adult, male, and that does not have a wife.

The scopes of the negative elements in these structures explain the FFG results. In a hierarchical structure as in a semantic reading, a negation of a node in the structure results in a disjunction made by attaching the negative element to alternate nodes below the initial negative element. The following schema illustrates this format:



The taller the structure and the higher up in it the negative element occurs, the more disjuncts or the more ORs you get. By simple calculation we see that the complexity relationships between explicit, morphological and definitional negatives that FFG found are exactly mirrored by considering the ST representations for sentences A, B, and C, taking into account the number of disjuncts generated by the negative elements in these structures. So a sentence like C, with the PDN "bachelor" gets no disjunction since the negative element is at the bottom of the semantic reading. Sentence B, with the morphological negative "unmarried" gets three disjuncts; and sentence A, with the explicit negative "not married" gets five disjuncts. Sentences with an explicit negative plus a morphological negative get fifteen disjuncts, and an explicit and a PDN get eight.

The following markers show that the relative complexities of the explicit, morphological, and implicit negatives also correspond to the FFG results.



Part 2: Three Connectedness/Relatedness Studies

Unlike the FFG experiment on negatives which dealt with the relative semantic complexity of sentence meanings, Gentner's recall experiments dealt with the relative connectedness of noun phrases in SVO sentences. Specifically, Gentner's experiment compared connectedness relations in such sentences for three different types of verbs: general verbs like "gave," connecting-specific verbs like "sold," and non-connecting specific verbs

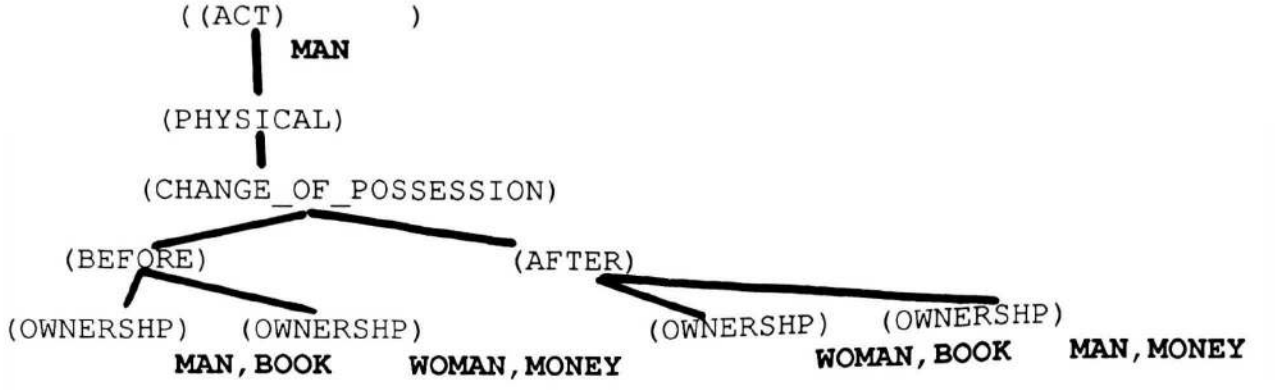
like "sent." When we compare the meanings of such verbs we see that a connecting specific verb like "sold," because it involves an exchange of merchandize and money between buyer and seller, provides for a greater degree of connectedness between noun phrases in its simple sentences than either the general or non-connecting specific verbs. While both "gave" and "sent" suggest a one-way transaction, "sent" is more specific or complex since it suggests a causative relation in the transaction, perhaps something like "cause to be given." One of Gentner's experiments involved reading simple sentences with these verbs to subjects and then asking them to recall the direct object, "woman" in the examples below, given the subject as a cue.

Basically, Gentner's approach was to distinguish between two hypotheses, both intended to test the definitional view. The first, the complexity hypothesis, is the one on which FFG had based their rejection of the psychological reality of definitions. The second Gentner called the connectivity hypothesis. Now, with respect to sentences like F, G, and H below the connectivity and complexity hypotheses make different predictions. The complexity hypothesis would predict that recall would be best for sentences with verbs like "gave" since the meanings of these verbs are the least complex and so leave more memory space available for recall. On the other hand, the connectivity hypothesis would predict that sentences with verbs like "sold" would bring about better recall since there is stronger connectedness between the referents of the noun phrases in such sentences.

In fact, it is the connectivity hypothesis that wins out in the Gentner results, with sentences with verbs like "sold" providing for much better recall of the object noun phrases than either sentences with verbs like "gave" or sentences with verbs like "sent." The definitional structures of ST do in fact provide a fairly straightforward explanation of these results in terms of the number of argument places in the ST representation of the verbs' meanings. Very simply, the more argument places there are in a structure the more connected the referents of the noun phrases that fill these argument places. As the following structures for sentences with the three verbs show, with arguments appearing in boldface, the sentence for "sold" has more work for the projection routine than either the sentence with "sent" or the one with "gave," which are equal in this respect. If we consider just the number of insertions of the subject noun phrase into its argument places we see that "sold" requires three, while "gave" and "sent" each require just two. If we consider the total number of argument insertions required for the subject, used as the cue in the Gentner experiment, and the direct object, which had to be recalled, then we see that while there are five argument insertions for "sold," there are three each for "gave" and "sent." This corresponds to the Gentner

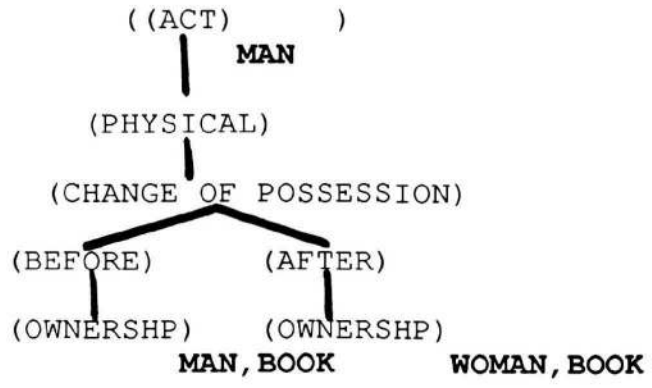
results.

F. The man sold the woman a book.

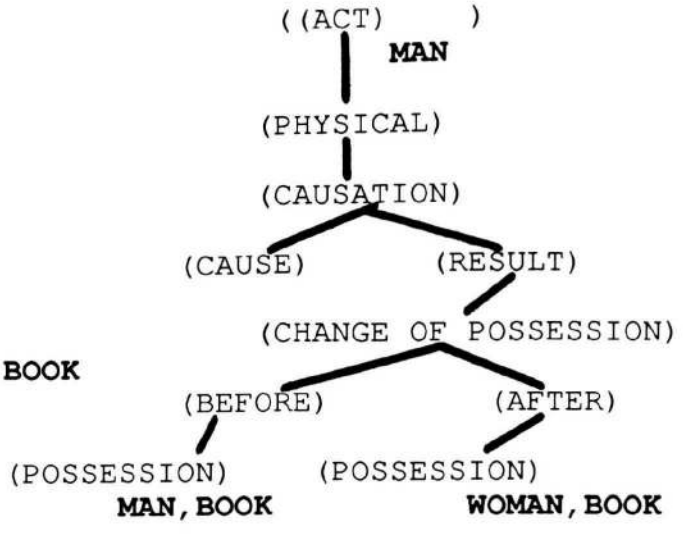


(G & H). The man _____ the woman a book.

G. gave



H. sent



Turning now to the FGWP evidence, we find that it concerns the relative degree of relatedness between noun phrases in simple sentences. Basically, FGWP claim that while on the basis of a supposedly definitional representation of causative verbs we would expect their subject and object NPs to be less related than the subject and object NPs of non-causative transitive verbs, in fact the experimental evidence shows that this is not the case. Now while FGWP choose to use the term "related" and Gentner, perhaps leaning towards the connectionist or parallel distributed processing approach, chooses to use the term "connected," and while FGWP employ a different methodology, direcly asking subjects to make relatedness ratings, the conclusions about greater or lesser relatedness or greater or lesser connectedness seem to be about the same thing. Very briefly, then, I would like

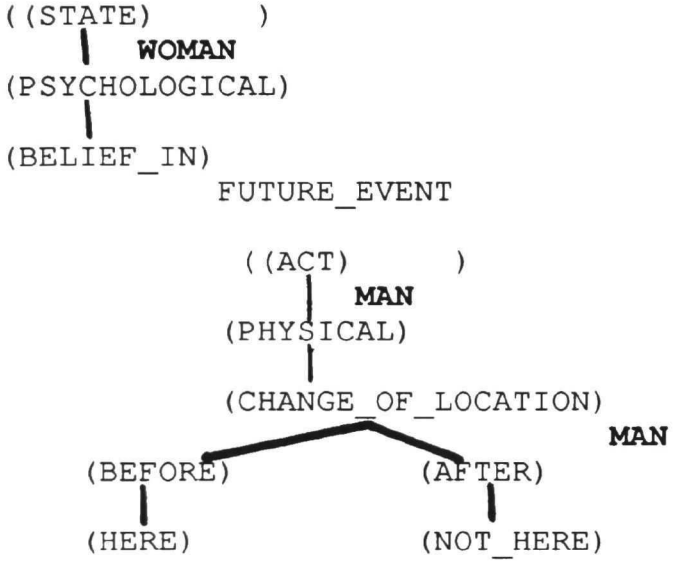
to show how the previous explanation of the Gentner evidence can also be applied to some of the FGWP results. Consider the following two sentences:

I. The woman expected the man to leave.

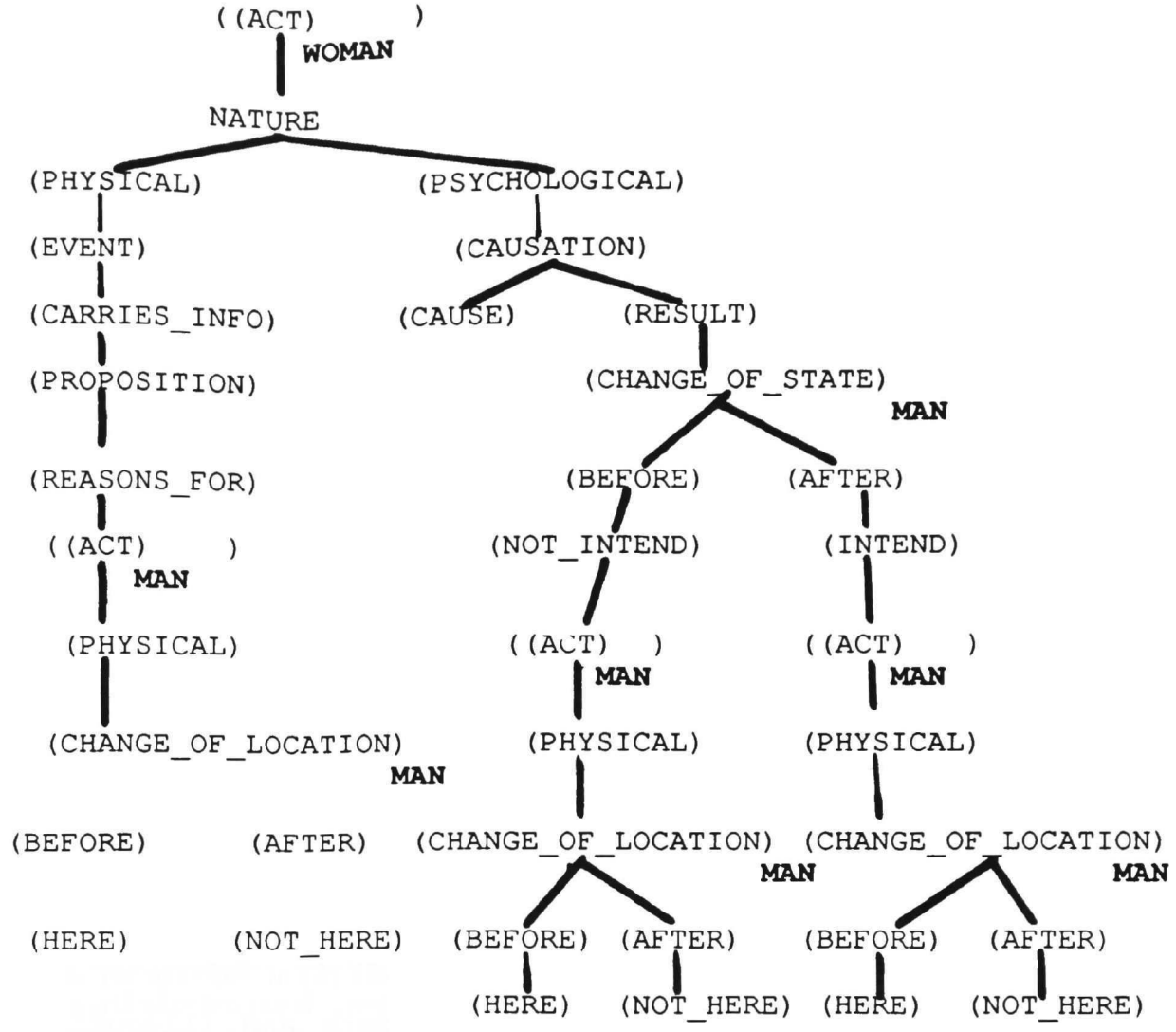
J. The woman persuaded the man to leave.

Gergely and Bever (1987) and FGWP both found that "the woman" and "the man" in (J) are judged to be more closely related than the same NPs in (I). FGWP offer a syntactic explanation for this which Gergely and Bever have shown not to work for other cases. In fact, Gergely and Bever suggest a semantic explanation: namely, that verbs like "persuade" whose meanings include a change in intention by the referent of the object NP brought about by the referent of the subject NP cause stronger relatedness judgements for these NPs than do other verbs which do not have this semantic structure. The Gergely and Bever study is very persuasive on this score. Nevertheless, consider the following representations of the meanings of these two sentences:

I. The woman expected the man to leave.



J. The woman persuaded the man to leave.



It is clear that the structure for "persuade" involves many more argument insertions than does the structure for "expect": eight for the former and only three for the latter. This, in my view, is what accounts for the difference in relatedness in the NPs in these sentences.

Moreover, this explanation would work for all verbs that involved a change of intention on the part of the referent of the object NP since all such verbs must have a branch in their semantic representations similar to the CHANGE-OF-STATE branch in the above reading for the "persuade" sentence, and this is the branch that has in it the proliferation of argument insertions below the highest BEFORE and AFTER nodes.

Conclusion: The explanations presented here for both the negative

sentences and the NP relatedness experiments are quite natural. But what seems compelling to me is that both of these explanations depend crucially on the definitional character of the representations. Moreover, the psychological data that these explanations handle so well arise both from opponents and exponents of the definitional approach. For all these reasons psycholinguists and specialists in cognitive science interested in the mental representation of word meaning should give more credence to the definitional view.

REFERENCES

Fodor, J. A., Garrett, M., Walker, E., & Parkes, C. (1980). Against Definitions. Cognition, 8, 263-367.

Fodor, J. D., Fodor, J. A., & Garrett, M. (1975). The psychological unreality of semantic representations. Linguistic Inquiry, 6, 515-531.

Fodor, J. A. (1981). Representations: Philosophical Essays on the foundations of cognitive science. Cambridge, MA: MIT Press.

Gentner, D. (1981). Verb semantic structures in memory for sentences: evidence for componential representation. Cognitive Psychology, 13, 56-83.

Gergely, G., & Bever, T. (1986). Relatedness intuitions and the mental representation of causative verbs in adults and children. Cognition, 23, 211-277.

Katz, J. (1972). Semantic Theory. New York: Harper and Row.

Katz, J. (1977). Propositional structure and illocutionary force. New York: Thomas Y. Crowell Co.

Katz, J. (1989). Cogitations: A Study of the Cogito in relation to the philosophy of Logic and Language and a Study of them in relation to the Cogito. New York: Oxford University Press.

Locke, J. (1975). An Essay Concerning Human Understanding. Ed. P. H. Nidditch. Oxford: Oxford University Press.

McNamara, T. P. and Miller, D. L. (1989). Attributes of theories of meaning. Psychological Bulletin, 106, no. 3, 355-376.

Lexical Cooccurrence Relations in Text Generation[1]

Leo Wanner
Projekt KOMET
GMD-IPSI
Dolivostr. 15
D-6100 Darmstadt
West Germany
(*email:* wanner@ipsi.darmstadt.gmd.dbp.de)

John A. Bateman
USC/Information Sciences Institute
4676 Admiralty Way
Marina del Rey
California 90292
U.S.A.
(*email:* bateman@isi.edu)

Abstract

In this paper we address the organization and use of the lexicon giving special consideration to how the salience of certain aspects of abstract semantic structure may be expressed. We propose an organization of the lexicon and its interaction with grammar and knowledge that makes extensive use of *lexical functions* from the Meaning-Text-Theory of Mel'čuk. We integrate this approach with the architecture of the PENMAN text generation system, showing some areas where that architecture is insufficient, and illustrating how the lexicon can provide functionally oriented guidance for the generation process.

1 Introduction

In this paper we address the organization of lexis[2] giving special consideration to the expression of the salience of certain aspects of abstract semantic structure — a set of phenomena which we call **perspectives** of that structure. These have been addressed rarely in approaches in generation so far: for example, [Jacobs 85] discusses the verbs "give" and "take" as two different expressions of the same event; and Iordanskaja *et al.* [Iordanskaja, Kittredge, and Polguère 88] propose an approach to linguistic paraphrasing by adapting the Meaning-Text-Theory (MTT) [Mel'čuk and Žholkovsky 70] and its paraphrasing rules. Here, we make more extensive use of the MTT in order to provide a richer organization of lexis and its interaction with grammar and knowledge than has been proposed previously. Moreover, we develop this approach in the context of a concrete generation environment, the PENMAN system [Mann and Matthiessen, 85], showing some areas where the existing architecture is insufficient and how the richer organization of lexis we propose can help.

The following set of examples gives an impression of the variety of linguistic phenomena that we include under the term perspective.[3] All the sentences can be interpreted as verbalizations of a single abstract semantic structure with differing aspects of that structure being given expression in each case. For example, in (3), the reader is made salient as a participant of the proposition; and in (5), a particular temporal aspect of the process, namely the beginning, is put into focus. While the variation that can be seen between (1) and (2) can already be treated in, for example, the current PENMAN system by exercising meaning options available in the grammar (i.e., (2) exhibits passivization), the variation shown in the remaining examples cannot be functionally motivated as possible realizations of the base form.

1. *"We use the adjective "electronic" to indicate that the dictionaries are deeply dedicated to computers."*
2. *"The adjective "electronic" is used to indicate that the dictionaries are deeply dedicated to computers."*
3. *"The reader gets an indication that the dictionaries are deeply dedicated to computers by the adjective "electronic"".*
4. *"By the use of the adjective "electronic" we illustrate the deep dedication of dictionaries to computers."*

[1]We would like to thank Elisabeth Maier, Hans Müller, Erich Steiner, and Elke Teich for fruitful discussions. John Bateman acknowledges the additional financial support of IPSI during the development of the ideas reported here.

[2]We use the term 'lexis' rather than 'lexicon' to cover both the static organization of lexical information and the dynamic aspect of the *use* of that information and its *interaction* with other components of the linguistic system. Lexis, in this sense, is a term we borrow from Systemic linguistics, cf. [Matthiessen 89].

[3]The basic sentence given under (1) is chosen from the introductory note of a text concerning the development of electronic dictionaries in Japan [EDR 88].

5. *"We create an indication that the dictionaries are deeply dedicated to computers by the adjective "electronic"*

Some of the phenomena running through these examples have been treated as **lexical cooccurrence** [Apresjan, Žholkovsky, and Mel'čuk 69] or **collocation** [Firth 51; Halliday 66; Hausmann 85]). Most extensively they are handled by I. Mel'čuk *et al.* in the scope of the Meaning-Text-Theory by means of **lexical functions** (LFs). Our approach to adding the ability to generate this range of variation under functional control by means of perspectives takes its starting point, therefore, from the notion of lexical cooccurrence defined by Mel'čuk. There is a large body of descriptive work based on the notion of LFs which has been carried out for different languages [Mel'čuk and Žholkovsky 84; Mel'čuk *et al.* 88; Žholkovsky 70; Reuther 78; Janus 71] and we will suggest how this body of knowledge can now provide significant input to work on text generation.

2 The Nature and Organization of Lexical Functions

Lexical cooccurrence in the scope of MTT is provided in terms of lexical functions which Mel'čuk defines as follows [Mel'čuk and Polguère 87]:

> *A lexical function* **f** *is a dependency that associates with a lexeme L, called the argument of* **f**, *another lexeme (or a set of (quasi-)synonymous lexemes) L' which expresses, with respect to L, a very abstract meaning (...) and plays a specific syntactic role. For instance, for a noun* **N** *denoting an action, the LF* $\mathbf{Oper}_1$ *specifies a verb (...) which takes as its grammatical subject the name of the agent of the said action and as its direct object, the lexeme* **N** *itself.*

The values for any particular application of a LF to a lexeme are provided by an **Explanatory Combinatorial Dictionary** (ECD); extensive dictionaries of this type for a number of languages have already been compiled by MTT researchers. Thus, for example, the ECD for English provides for the following applications of the LF $\mathbf{Oper}_1$: Oper_1 ("influence") = "exert", Oper_1 ("punishment") = "administer". These give lexical verbs appropriate for use when the argument is to appear as a direct object to form a combination where an agent (optionally) acts upon some patient; e.g.: *He exerted influence on P...*, *He administered a punishment...*, but not, *he exerted a punishment...*, *he administered an influence...*.

Cooccurrence relations of this kind are pervasive in natural language and need to be captured in the representation of a language's lexical resources. Such co-occurrence relations can be rather arbitrary and so are unlikely to be supportable by, for example, distinctions maintained in the knowledge base. Their *meaning* is not, however, arbitrary. An important claim of MTT is that each LF represents a particular abstract meaning which remains invariant across its various applications. Thus, for example, further LFs include: $\mathbf{Func}_0$ with the meaning 'something takes place' (Func_0 ("accident") = {"occur", "happen"}, as in the sentence: *"The accident occured two hours ago."*); **Result** standing for a state following the process addressed (Result ("subject") = "master", as in the sentence: *"John mastered his subject."*); and **Liqu** expressing an active process termination. This latter LF is often used in so called *composed* LFs where a number of LFs are combined in a predefined order ($\mathrm{LiquFunc}_0$ ("Fire") = {"extinguish", "put out"}, as in the sentence: *"The fire brigade could put out the fire quickly."*).

LFs typically correspond to knowledge at varying levels of abstraction in addition to lexical information, these classes are still very heterogenous. Previous approaches that have made use of LFs in generation (e.g., [Kittredge and Mel'čuk 83; Iordanskaja, Kittredge, and Polguère 88; Bourbeau *et al.* 89]) have been hindered by this. Work in progress at IPSI suggests that the large number of heterogeneous LFs used within MTT can be organized coherently in terms of the functions and semantic distinctions that they represent. Based on this, we have defined part of a general model of lexis with a taxonomic organization underlying it, within which the most general structures provide the representations of lexical semantics and the most delicate ones lexicalization. For the purposes of this paper, we will restrict attention to the organization of LFs that are particularly relevant for modeling situation perspectives as illustrated in our examples above. In Figure 1 we set out in network form the more general distinctions in meaning that the LFs we discuss

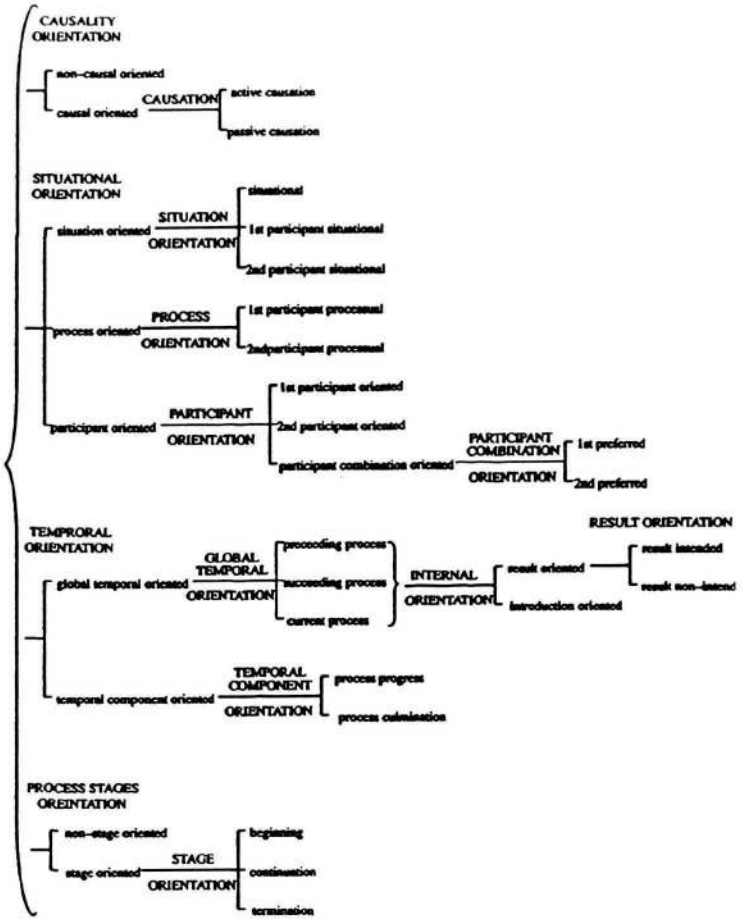


Figure 1: The hierarchical organization of lexical functions in network form

here cover.[4] The network explicates LFs by classifying each of them according to a particular set of semantic features. The general function of the network is thus to relate particular LFs to the functional conditions for their application. This defines the meaning that any LF expresses and so provides a functionally organized key into the LF-oriented dictionaries being developed within MTT. The network also shows the hierarchial arrangement lying behind the meaning of LFs and so reflects the relation of perspectives to one another.

We will now briefly describe in semantic terms a representative set of the LFs covered by the network, showing how the network relates perspectival presentation decisions to choices of LFs.[5] Then, with the organizational network of perspectives in place and motivated, we show how it can

[4]The notation of Figure 1 follows that used within the NIGEL grammar of the PENMAN system for the specification of grammar possibilities. Names in capitals represent the names of choice points, and names in lower case features which may be selected: one from each choice point; also square brackets represent disjunction of features and braces conjunction. Such networks can be readily expressed in a number of distinct formalisms, e.g., FUG (cf. [Kasper 88]), LOOM (cf. [Kasper, 89]).

[5]In the full version of this network, the consequences of each possible selection of features for LF selection is specified; space precludes a detailed discussion at this point, although examples are given below.

be used to guide the generation process to produce the kinds of variation illustrated in (1)-(5).

§2.1 Situation introduction

When a situation is introduced, this may be done respecting a number of varying types of salience — e.g., the salience of particular participants of a situation or the situation itself. Mel'čuk characterizes abstract situations by **key terms** which on the syntactic level are realized as nominals (or, more specifically, in most cases, as nominalized verbs) and their participants; the key term is designated by the LF S_0, the participants as S_i (i-th participant of the situation). Thus, looking at the situation of "teaching", the ECD for English offers us: S_0 ("teaching") ="teaching", S_1 ("teaching") = "teacher", S_2 ("teaching") = "pupil".

The selection of particular combinations of process and participants according to differing attributions of salience is then provided in the scope of the ECD by LFs of the groups **Func**, which stands for the salience of the lexeme labeling the situation, **Oper**, which stands for the salience of one of the participants, and **Labor** which stands for the salience of a combination of the participants. The selection of these broad groups is made in the network by the choices available under SITUATIONAL ORIENTATION, by the features 'situation oriented' (Func) and 'participant oriented' (Oper, Labor).

These are further differentiated according to which participants are affected; e.g.: Oper_1 makes the 'first' participant of the situation salient (i.e., the participant for which the LF S_1 provides a lexeme) and Oper_2 the 'second' (i.e., the participant for which the LF S_2 provides a lexeme): Oper_2 ("influence") = "be under". Similarly, Func_0 makes the key term of the situation itself salient, while Func_1 introduces the situation with particular respect to the first participant: Func_0 ("problem") = "exist", Func_1 ("problem") = "come [from]". Labor_{12} makes the first and the second participant salient, the first more then the second, Labor_{21}, on the contrary, makes the second participant more salient, e.g.: Labor_{12} ("authority") = "vest [with]", Labor_{21} ("authority") = "owe [to]". These options are controlled by the further selections of participants to be accorded salience in the choice points SITUATION ORIENTATION and PARTICIPANT ORIENTATION.

Finally, the third option in the SITUATIONAL ORIENTATION system, 'process orientation' is responsible for the neutral LF V_0, which provides the most direct lexical verb for realizing the key term of a situation; e.g., V_0 ("influence") = "[to] influence".

§2.2 Temporal dependency

LFs also address the global arrangement of a process on the temporal axes by the definition of its preceding and succeeding processes. These considerations are reached in the network by a feature selection of {global temporal oriented, ..., } from the alternatives of the TEMPORAL ORIENTATION choice point. These alternatives call for the application of the LFs **Prox** and **Result**; examples of which from the ECD for English are: ProxFunc_0 ("storm") = "brew", ResultFunc_0 ("storm") = "subside". In addition, the internal temporal aspects of a process, represented by its stages, are reflected by the corresponding triple of "phasal" LFs: **Incep** for the beginning, **Cont** for continuing, and **Fin** for the termination stage. These meanings are reached via the features under the 'stage oriented' option in the choice point PROCESS STAGES ORIENTATION in the network.

§2.3 Results and consequences

Situations can also be expressed so as to give salience to their *results*. The treatment of this requires a consideration of the *intended* result of the situation — the actual LF chosen depends on whether that result was achieved or not. These options are found under the choice point INTERNAL ORIENTATION and RESULT ORIENTATION. If the result of the process was the intended result (i.e., the 'purpose' of the carrying out the process), then the $\textbf{Real}_i$, $\textbf{Labreal}_{ij}$, and $\textbf{Fact}_i$ groups of LFs are applicable; in the opposite case, the $\textbf{AntiReal}_i$, $\textbf{AntiLabreal}_{ij}$, and $\textbf{AntiFact}_i$ groups apply. Each of these groups provide further the salience either of the key term of the situation itself or of the various participants of the situation as determined by the simultaneous selections of features made under (in this sense Real_i and AntiReal_i correspond to Oper_i, Labreal_{ij} and AntiLabreal_{ij} to Labor_{ij}, and Fact_i and AntiFact_i to Func_i). For example: Real_2 ("order") = "obey", AntiReal_2 ("order") = "refuse".

§2.4 Causality

Situations can also be expressed so as to make the *causality* relationships that the situation enters into explicit or not; these options are considered by the choice point CAUSALITY ORIENTATION, which is responsible for application of either the LF **Caus** or **Perm**.

- The Caus function provides the active causer of the situation, as in the case of "problem"; e.g., CausFunc_0 ("problem") = "pose".
- Perm presupposes a 'permission', or allowance or acceptance, of the occurrence from the agent; e.g., PermFunc_0 (problem) = "tolerate".

3 Guiding the Generation Process by Lexis

The concrete generation system in which we are realizing the ideas proposed in this paper is the PENMAN system [Mann and Matthiessen 85]. The linguistic core of PENMAN is a large systemic-functional grammar of English, the NIGEL grammar [Matthiessen 83]. The semantic interface of NIGEL is defined by a set of *inquiries* mediating the flow of information between the grammar and external sources of information. PENMAN provides structure for some of these external sources of information, including a conceptual hierarchy of relations and entities, called the *Upper Model* (UM) [Bateman, Kasper, Moore and Whitney 89]; the UM is used as an interface between the organizational structures of the Domain Knowledge (DK) and the grammar's inquiries. PENMAN accepts demands for text to be generated in the notation of the PENMAN *Sentence Plan Language* (SPL) [Kasper 89]. SPL expressions are lists of terms describing the types of entities and the particular features of those entities that are to be expressed in English. The features of SPL terms are either semantic relations to be expressed, which are drawn from the upper model or from domain concepts subordinated to the upper model, or direct specifications of responses to NIGEL's inquiries.[6]

To generate any of the sentences (1)-(5) above using PENMAN, therefore, we must define appropriate SPL input. However, as mentioned in Section 1, these input specifications do not, at present, capture the generalization that these sentences share significant aspects of their meaning. To capture this, while still maintaining complete functional control of the generator, we introduce a more abstract input specification, from which particular SPL specifications are constructed depending on additional salience-oriented semantic distinctions. These semantic distinctions are specified in terms of the hierarchical organization of the meanings of LFs shown in the network of Figure 1. This organization provides a decision network representing the perspectives available and the functional motivations for choosing one perspective over another. Each of the decision points in this network may place constraints on the mapping between the abstract input level and SPL. These decisions themselves need to be made by a text planning component — the network represents the *capability* of generating variation under control rather than the control process itself. In this sense, lexis as the stratum containing perspective information provides a controlling mechanism for the generation process entirely analogously to the grammatical network defined by NIGEL.

4 Example of perspective-guided generation

We now illustrate the realization of some chosen perspectives in detail. Consider the clauses (1), (3), (4), and (5) given in Section 1. The SPL input specifications necessary to generate each of these clauses are set out in Figure 2.[7] As we can see, there is no connection between these since the generalization that they refer to the same situation is captured neither within the grammar, nor the upper model. Our new level of abstract input to the generation process, which corresponds more with Mel'čuk's conception of 'abstract situation' introduced in Section 2, provides this connection as follows. Abstract situations are represented in terms of a general type and a set of participants drawn from the lexemes defined with respect to the Domain Knowledge; for example, the abstract input for the situation underlying sentences (1), (3), (4), and (5) may be set out thus:[8]

[6] For full details of the PENMAN system and its components, see the PENMAN documentation [The PENMAN Project 89].

[7] Note that in this figure, in order to save space, we share the varables `we`, `N1`, `A1`, `AS1` across the distinct SPL specifications; this would not normally be done.

[8] The notation $\uparrow S_i$ is used to indicate that the value given is *not* the value of the LF S_i itself, it is rather the value of the role that the LF delivers; i.e., S_1 ("use") is "user".

```
(C1 / use
    :actor (we / person)
    :actee (N1 / adjective
               :name electronic)
    :purpose (A1 /indicate
                 :actor we
                 :subject-matter
                     (AS1 / dedicate
                          :domain (dictionaries / thing)
                          :range (computers / thing)
                          :manner (deep / sense-and-measure-quality))))

                          SPL specification for sentence (1)

(C2 / get :actor (r / reader) :means N1 :actee A1)

                          SPL specification for sentence (3)

(C3 / illustrate :actor we :actee AS1 :means N1)

                          SPL specification for sentence (4)

(C4 / create :actor we :actee A1 :means N1)

                          SPL specification for sentence (5)
```

Figure 2: SPL specifications for differing perspectives on a situation

$$\left[\begin{array}{lll} S_0 & use & \\ \uparrow S_1 & we & \\ \uparrow S_2 & adjective\ \text{`electronic'} & \\ \uparrow S_3 & \left[\begin{array}{lll} S_0 & indication & \\ \uparrow S_2 & reader & \\ \uparrow S_3 & \left[\begin{array}{lll} S_0 & \text{(deep) } dedication & \\ \uparrow S_1 & dictionaries & \\ \uparrow S_2 & computers & \end{array}\right] & \end{array}\right] & \end{array}\right]$$

In order to generate sentences from this specification, we need to construct appropriate SPL expressions. This we achieve by following the semantic alternatives made in the LF network of Figure 1, applying the constraints that it specifies to compose a mapping between the abstract input and SPL.

Thus, for example, consider the context of use where a text planner has determined, in addition to expressing the situation shown in the abstract input, that that situation is to be presented textually as one in which the process is introduced neutrally, without respect for what preceded or succeeded, and with the process and the first participant (*we*: S_1) made relatively more salient. This corresponds to the set of LF network features {non-causal oriented, non-stage oriented, global temporal oriented, current process, introduction oriented, process-oriented, 1st participant processual}. This set of features governs the selection of the LF V_0, which is applied to the key-term of the situation, i.e., to S_0 of the input form: the lexeme associated with the DK concept *use*. The ECD for the language then supplies a candidate lexical item — in this case, the process "use".

We integrate the information from the ECD by requiring lexical items to be linked to concepts which are subordinated to the PENMAN upper model. It is then possible to determine, by inheritance, the particular set of upper model/semantic role relations that are appropriate for a process of any type. The concept for "use" is classified as a *nondirected-action* in the upper model and so the role-set `:actor, :actee` is inherited. The fillers of these roles are then selected from the ordered set of participants specified in the abstract input under S_1, S_2. The process then recurses for the complex filler of S_3 — filling, in this case, `:purpose` upper model relation — and the SPL given in Figure 2 for sentence (1) is constructed.[9]

[9]The association of the abstract situational roles S_i and the roles drawn from the upper model in fact offers

If the text planning component had determined that a different set of presentational LF features were necessary, then a different LF would be selected for application to the key-term of the abstract input. Thus, with the selection of the features {non-causal oriented, non-stage oriented, global temporal oriented, succeeding process, introduction oriented, participant oriented, 1st participant oriented}, which expresses the effect of the process *use* with salience on its first participant, the LF ResultOper_1 is selected and, here, the ECD gives the process *illustrate*. This term is then, again, selected as the main term in the corresponding SPL specification and, as before, since it is also linked into the upper model, we know that the relevant role set is `:actor, :actee, :means`. The further mapping of situational roles S_i to available UM-roles then provides the necessary fillers for the slots in the SPL. This gives the SPL for sentence (4).

In sentences (3) and (5), the interaction between the lexical network and the situation subordinated under S_3 in the abstract input is shown.[10] For the situation of 'indication', then, when the LF features: {non-causal oriented, non-stage oriented, global temporal oriented, current process, introduction oriented, participant oriented, 2nd participant oriented} are required, expressing that the situation is introduced with emphasis on its internal composition and participants and that the second of those participants is the more salient, then the LF Oper_2 is selected for application to the filler of $\uparrow S_3$ (i.e., "indication"). The ECD in this case supplies "get". Note that here, the LF Oper_2 also has consequences for the latter mapping between situational roles and upper model roles; the key-term itself, S_0, is now associated with the role `:actee`. This provides the SPL specification for sentence (3).

Finally, with the selection of LF features {non-causal oriented, stage oriented, beginning, participant oriented, 1st participant oriented, global temporal oriented, current process, introduction oriented}, the LF IncepOper_1 is selected. When this is applied to "indication", the ECD gives the process "create" and the SPL for sentence (5) is set up accordingly.

5 Conclusion

We have shown how lexical cooccurrence relations can be used to express the salience of particular aspects of abstract semantic structures and how they can be organized to influence the generation process. A specification of perspectival presentatation features as defined in the network of Figure 1 makes it possible to generate rather varied surface realizational forms. We can view this network as a candidate for the *textual organization* of lexis — which complements the more traditional 'propositional' organization found in lexical discrimination nets (e.g., Goldman, 1975) and thesauri. The functional meanings of LFs we propose, although arguably inherent in the MTM, have not formerly been extracted as an explicit principle of organization. We suggest that this kind of organization may substantially enhance the information collected by MTM researchers. Finally, although we have restricted ourselves in this paper to details that are particularly relevant for modeling situation perspectives, we are working at a general model of lexis including, e.g., a semantically motivated classification of verbs, relations, etc. For this we also use a set of further LFs represented on various levels of abstraction.

References

[Apresjan, Žholkovsky, and Mel'čuk 69] Yu.D. Apresjan, A.K. Žholkovsky, I.A. Mel'čuk. 'On a Possible Method of Describing Restricted Lexical Cooccurrence', In *Russkij Jazyk v Nacionalnoj Shcole*, 6, 61-72.

[Bateman and Paris 89] J.A. Bateman, C.L. Paris "Phrasing a text in terms the user can understand", In *Proceedings of IJCAI 89.*

[Bateman, Kasper, Moore, and Whitney 89] J.A. Bateman, R.T. Kasper, J.D. Moore, R. Whitney. "The PENMAN Upper Model — 1989", ISI Research Report, USC/Information Sciences Institute, Marina del Rey, CA

another significant source of presentation variability which may also be addressed in terms of LFs. We do not discuss this further within the confines of the present paper however.

[10] Work elsewhere (e.g., [Bateman and Paris 89]) has shown that *propositionally* embedded components of an input specification can be linguistically realized under certain textual conditions as unembedded, or as dominating, constituents. This is the case here, although space precludes a more thorough discussion.

[Bourbeau *et al.* 89] L. Bourbeau, D. Carcagno, K. Kittredge, and A. Polguère "Text Synthesis for Marine Weather Forecast", Final Report, Odyssey Research Associates Inc., Montréal

[EDR 88] "Electronic Dictionary Project", Technical Report, Japan Electronic Dictionary Research Institute, Ltd.

[Firth 51]J.R. Firth "Modes of Meaning", In *Papers in Linguistics 1934-51*, London: Longman, pp190-215.

[Goldman 75] N. Goldman. "Conceptual Generation". In R.C. Schank (ed.) *Conceptual Information Processing*, North-Holland Pub.Co., Amsterdam.

[Halliday 66] M.A.K. Halliday "Lexis as a linguistic level", in C.E. Bazell *et al.* (eds.) *In Memory of J.R. Firth*, London: Longman, pp148-162.

[Hausmann 85] F.J. Hausmann. "Kollokationen im deutschen Wörterbuch. Ein Beitrag zur Theorie des lexikographischen Beispiels", in H. Bergenholtz, J. Mugdan (eds.) *Lexikographie und Grammatik, Akten des Essener Kolloquiums zur Grammatik im Wörterbuch*, pp118-129.

[Iordanskaja, Kittredge, and Polguère 88] L. Iordanskaja, R. Kittredge, A. Polguère "Lexical Selection and Paraphrase in a Meaning-Text Model", Paper presented at the 4th International Workshop on Natural Language Generation, St. Catalina, CA. To appear in: *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, Cecile L. Paris, William R. Swartout and William C. Mann (eds.), Kluwer Academic Publishers.

[Jacobs 85] P.S. Jacobs. "A Knowledge-Based Approach to Language Production", Report No. UCB/CSD 86/254, Univ. of California at Berkeley

[Janus 71] E. Janus. "Five Polish Dictionary Entries...", In *Naučno-techničeskaja informacia*, 2.11, 21-24.

[Kasper 88] R.T. Kasper "An Experimental Parser for Systemic Grammars", in *Proceedings COLING '88*

[Kasper 89] R.T. Kasper. "A Flexible Interface for Linking Applications to PENMAN's Sentence Generator", in *Proceedings of the DARPA Speech and Natural Language Workshop*, Philadelphia, 1989.

[Kittredge and Mel'čuk 83] R. Kittredge, I.A. Mel'čuk "Towards a computable model of meaning-text relations within a natural sublanguage", in *Proceedings of the IJCAI 1983.*

[Mann 85] W.C. Mann. "An Introduction to the NIGEL text generation grammar", in J. Benson, W. Greaves (eds.) *Systemic Perspectives on Discourse, Vol. I, Selected Theoretical Papers from the 9th International Systemic Workshop 1983*, NJ, Ablex, 1985

[Mann and Matthiessen 85] W.C. Mann, C.M.I.M Matthiessen. "A demonstration of the NIGEL text generation computer program", in J.D. Benson, W.S. Greaves (eds.) *Systemic Perspectives on Discourse, Vol. I, Selected Theoretical Papers from the 9th International Systemic Workshop 1983*, NJ, Ablex, pp. 50-83, 1985

[Matthiessen 83] C.M.I.M Matthiessen. "Systemic grammar in computation: the NIGEL case", in *Proceedings of the First Conference of teh European Chapter of the Association for Computational Linguistics*, Pisa, Italy, 1-2 September, 1983.

[Matthiessen 89] C.M.I.M Matthiessen. "Lexico(grammatical) Choice in Text Generation", revised version of the Paper presented at the 4th International Workshop on Natural Language Generation, St. Catalina, CA, 1988. To appear in: *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, Cecile L. Paris, William R. Swartout and William C. Mann (eds.), Kluwer Academic Publishers.

[Mel'čuk and Polguère 87] I.A. Mel'čuk, A. Polguère. "A Formal Lexicon in the Meaning-Text Theory (or How to Do Lexica with Words)", *Computational Linguistics*, 13.3-4, 276-289.

[Mel'čuk and Žholkovsky 70] I.A. Mel'čuk, A.K. Žholkovsky. "Towards a Functioning Meaning-Text Model of Language", *Linguistics*, 57, pp. 10-47

[Mel'čuk and Žholkovsky 84] I.A. Mel'čuk, A.K. Žholkovsky. *Explanatory Combinatorial Dictionary of Modern Russian*, Wiener Slawistischer Almanach, Vienna 1984.

[Mel'čuk *et al.* 88]Mel'čuk, N. Arbatchewsky-Jumarie, L. Elnitsky, A. Lessard. *Dictionnaire explicatif et combinatoire du francais contemporain*, Presses de l'Université de Montréal, Montréal.

[Reuther 78] T. Reuther "Plädoyer für das Wörterbuch",, *Linguistische Berichte*, 57, 25-48.

[The PENMAN Project 89] "The PENMAN Documentation: Primer, User Guide, Reference Manual, and NIGEL Manual", USC/Information Sciences Institute, Marina del Rey, California.

[Žholkovsky 70] Žholkovsky. "Materials for a Russian-Somali Dictionary", *Mashinnij perevod i prikladnaja lingvistika*, 13, 35-63.

Learning Lexical Knowledge in Context: Experiments with Recurrent Feed Forward Networks

Steven L. Small

Department of Neurology
University of Pittsburgh

Abstract

Recent work on representation in simple recursive feed forward connectionist networks suggests that a computational device can learn linguistic behaviors without any explicit representation of linguistic knowledge in the form of rules, facts, or procedures. This paper presents an extension of these methods to the study of lexical ambiguity resolution and semantic parsing. Five specific hypotheses are discussed regarding network architectures for lexical ambiguity resolution and the nature of their performance: (1) A simple recurrent feed forward network using back propagation can learn to predict correctly the object of ambiguous verb "take out" in specific contexts; (2) Such a network can likewise predict a pronoun of the correct gender in the appropriate contexts; (3) The effect of specific contextual features increases with their proximity to the ambiguous word or words; (4) The training of hidden recurrent networks for lexical ambiguity resolution improves significantly when the input consists of two words rather than a single word; and (5) The principal components of the hidden units in the trained networks reflect an internal representation of linguistic knowledge. Experimental results supporting these hypotheses are presented, including analysis of network performance and acquired representations. The paper concludes with a discussion of the work in terms of computational neuropsychology, with potential impact on clinical and basic neuroscience.

1. Introduction

Connectionist approaches to the study of language, vision, and memory have led to altered perspectives on the nature of cognition [Churchland and Sejnowski, 1987]. In particular, this work has meant the rethinking of the computer metaphor for the mind, such that human memory does not necessarily have to be a "place" to "store" information and human knowledge can be more than "facts" and "inference rules". These computational concepts can be rejected (or at least questioned) without giving up the metaphor of human mental computation [Feldman, 1989].

Parsing has always played a prominent role in the computational study of human language. One reason for this, of course, was the engineering importance of parsing to early researchers in artificial intelligence; their goal was to access newly devised information resources using "natural" language. Cognitive scientists have also focused on parsing issues, with connectionist approaches contributing increasingly to this attention [Cottrell and Small, 1983; Waltz and Pollack, 1985].

Recent work on linguistic representation in connectionist models [Elman, 1989] has profound significance for the psycholinguistic and computational study of human language. In this work, Elman constructs a feed forward connectionist network [Rumelhart, et al., 1985] with only one (easily implemented) recurrent structure [Cottrell and Fu-Sheng, 1989], and trains it to analyze sentences. For each word presented in a particular sequence, the network must predict the next word expected. When the training has been completed, the network has acquired the ability to perform the desired task. Furthermore, in analyzing (statistically) the nature of the acquired "knowledge", Elman found that similar words, both semantically and syntactically, clustered together. He subsequently used his technique to build a network to learn to predict subject/verb agreements in sentences with relative clauses (differing in number).

This work succeeds for the first time at accomplishing a task that has been attempted a number of times in the recent history of cognitive science [Small and Rieger, 1982] without as much success. This work demonstrates (to a limited, but not insignificant

Full Address: Department of Neurology, University of Pittsburgh, 325 Scaife Hall, Pittsburgh, PA 15261. Phone: (412) 648-9200. Network: small@cadre.dsl.pitt.edu.

degree) that a computational device can learn linguistic behaviors without any explicit representation of linguistic knowledge in the form of rules, facts, procedures, or other symbolic schemes. Furthermore, it does so using a simulation technique that has much closer analogies to the human neurobiological substrate (i.e., neurons and their connections) than do any symbol processing approaches. Of course, there are great differences between connectionist networks (of all kinds) and biological neural "networks", and there are important contributions to cognitive scientific knowledge available from the study of logical representations. However, the ability of a simple network of impoverished computational units to acquire linguistic knowledge is important.

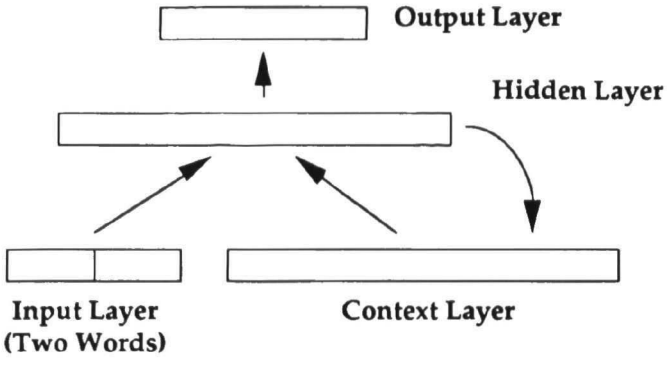


Figure 1: Hidden Recurrent Net with Double Input Buffer

2. Lexical Ambiguity Resolution

Representation of semantic and pragmatic context for lexical disambiguation has been a difficult problem, especially in contexts involving more than one sentence. In the work described here, a simple recurrent feed foward network (as in Figure 1) was trained to predict the next word in a sequence of lexical inputs. The correct desired output word depends on the semantic context of the previous sentence. A set of two sentence "stories" constituted the basic input to the system. Sentence 1a is an example "story" of this type.

(1a) "A man fights. He takes out the assailant."

As part of the experimental method, two simplifying assumptions were made initially, one of which was subsequently lifted. Throughout all of the experiments, the sentences have been simplified by removal of the articles. Sentence 1b shows the example story in the form actually used.

(1b) "Man fights. He takes out assailant."

A further simplification of the experimental method is the merging of "take" and "out" into a single word "take-out". Note that this was done in some but not all of the experiments, and constitutes an interesting part of the experimental design. Sentence 1c shows this input form of the example story.

(1c) "Man fights. He takes-out assailant."

This change simplifies the problem by (a) decreasing the number of total input words and thus the width of the input vector; and (b) decreasing the distance between the contextually important antecedent word and the ultimate ambiguity resolution task word (i.e., predicting the next word after "take out").

The input to the system on any experimental trial included sequences of three or four stories. On each trial, some textual feature or computational parameter was investigated. The two experimental endpoints consisted of the ability of the network to learn the task (i.e., convergence) and the number of trials needed to learn the task. Text comprehension features investigated included the following:

(a) Pronominal gender agreement;
(b) Equivalent contexts for different objects;
(c) Different contexts for the same objects;
(d) Distance between contextual prime and ambiguous word; and
(e) Size of the input buffer.

Computational manipulations were also studied, and aspects that could affect convergence included:

(a) Network learning parameters;
(b) Training instance presentation; and
(c) Network architecture.

The ability of the networks to find solutions to the problems presented suggest a number of things about human linguistic representations, as noted by Elman [1989]. In addition, the ways in which these networks are manipulated to effectuate or improve learning may have implications for language teaching, especially in second language learning or in language learning following damage to the nervous system.

3. Hypotheses and Experiments

All experiments were conducted with feed forward networks and learning by back propagation of error [Rumelhart, et al., 1985].

Five hypotheses motivated the work:

(1) A simple recurrent feed forward network using back propagation can learn to predict correctly the object of ambiguous verb "take out" in specific contexts;

(2) Such a network can likewise predict a pronoun of the correct gender in the appropriate contexts;

(3) The effect of specific contextual features increases with their proximity to the ambiguous word or words;

(4) The training of hidden recurrent networks for lexical ambiguity resolution improves significantly when the input consists of two words rather than a single word; and

(5) The principal components of the hidden units in the trained networks reflect an internal representation of linguistic knowledge.

The experiments were done with networks of three layers, using simple recurrence of hidden units, as shown in Figure 1. The distinct input words were each encoded as if they were orthogonal vectors in an n-dimensional space, where n is the total number of words in the trial (i.e., with six distinct input words, the first one would be encoded as 000001, the second as 000010, and so forth). When the input consists of more than one word (e.g., two words), each word is encoded as before, but with multiple buffer positions, each one consisting of the vector for one word (e.g., the input vector width for inputs of two words becomes 2n). Outputs are encoded separately, with one bit position for each possible output item (i.e., words, concept representations, or case frame data).

The use of one hidden layer (rather than two or three) and the ideal number of hidden units (generally 3 to 4 times the number of coded input units) were empirically determined through many experimental trials. The learning rate (nu or epsilon) was generally kept at 0.6 and the momentum (alpha) at 1.0. Changes in these values had little effect on convergence of the experimental network configurations. No attempt was made to minimize convergence time, and convergence was defined as achieving the correct binary output values for all inputs, with a value < 0.4 defined as zero, a value > 0.6 defined as one, and anything in between undefined, as per the suggestions of Fahlman [1988].

Experiments were conducted with three or four sentence pairs (which we call "stories"), along the lines of those shown above. The training sets presented the input data one word at a time (input buffer size = 1) or two words at a time (input buffer size = 2). Examples of both training instance types for Sentence 1b are shown in Figure 2. Example stories to study both ambiguity resolution and pronoun gender agreement are shown below. These sentences (2a-d) were presented in several different ways during the experiments. The presentations involved either (a) the first three stories or all four stories; (b) a single "take-out" word or two separate words; and (c) either one input word at a time or two input words at a time (as seen in the example training set of Figure 2).

(2a) "Man fights. He takes out assailant."
(2b) "Woman cleans. She takes out garbage."
(2c) "Man loves. He takes out licence."
(2d) "Woman eats. She takes out supper."

INPUT	OUTPUT
man	fights
fights	*
*	he
he	takes
takes	out
out	assailant
assailant	*

Single Word Input Training Set

INPUT BUFFER #1	INPUT BUFFER #2	OUTPUT WORD
man	fights	*
fights	*	he
*	he	takes
he	takes	out
takes	out	assailant
out	assailant	*

Two Word Input Training Set

Figure 2: Example Training Sets with One or Two Input Words

Note that capital letters are not represented, but end of sentence periods are included (as the asterisk in the example training set of Figure 2).

4. Experimental Results

Approximately one hundred experiments were conducted, and some general conclusions are possible on the basis of what was learned empirically from those studies. Fourteen experiments, restricted to three and four story sequences, are summarized in Table 1, and labelled Experiments 1-14. Several parameters that were not varied in these fourteen experiments are not listed in the table, including the number of hidden layers (1), the learning rate (0.6), and the momentum (1.0).

The information included in the table consists of the following: The input buffer width is the number of vectors, each representing a single word, that were input to the network; the networks were presented with either one or two word inputs. The words "take out" were represented in some experiments as a single word "take-out" and in others as two separate words. The input of a "clear signal" (a vector of all zeros) after each epoch aided convergence, as per the empirical observations of Blumenfeld [1989]. The hidden layer fraction is the ratio of hidden layer width to input layer width (before recurrence). A network was considered to converge if it produced the correct results for the training set. This was always true when the mean squared error of the network was less than 0.1. A network was considered to be monotonic when its mean squared error never increased during training. The number of trials shown consists of epochs (complete presentations of the training set).

The hypotheses enumerated above proved to be mostly correct. It was possible to construct feed forward hidden recursive networks to predict the object of the verb "take out" in context (Hypothesis 1). Experiments using the same number of contexts (e.g., "fights") as ambiguous verbs (e.g., "takes out" meaning "knock out with a punch") converged the most readily (these experiments are shown in Table 1). Experiments with more contexts than ambiguous verbs also converged, but not as readily. Experiments with a greater number of ambiguous verbs than distinct contexts did not converge. Experiments including both male and female agents converged more readily than did experiments in which all the stories contained male agents only. The networks were not only able to predict the correct pronoun in context (Hypothesis 2), but actually improved their performance by having this additional nonredundant element of context to use in forming their internal (hidden unit vector) encodings.

By using "take out" as two words in some experiments but as a single word in others, the distance between the contextually relevant antecedent word and the ambiguous word was varied. Better convergence was obtained when it was encoded as one

EXPERIMENT #	1	2	3	4	5	6	7	8	9	10	11	12	13	14
PARAMETERS														
# of stories	3	3	3	3	4	4	4	4	4	4	4	4	4	4
# of "take out" words	1	1	2	2	1	1	1	2	2	2	2	2	2	2
Input buffer width	1	1	1	1	1	1	1	1	1	2	2	2	2	2
Genders represented	M	M	M	M	M	M	M/F	M	M/F	M/F	M/F	M	M	M
# of priming verbs	3	3	3	3	4	4	4	4	4	4	4	4	4	4
# of direct objects	3	3	3	3	4	4	4	4	4	4	4	4	4	4
Clear each epoch?	N	Y	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	Y
Input width	10	10	11	11	12	12	14	13	15	30	30	26	26	26
Hidden width	40	40	44	44	48	48	56	52	60	45	45	39	65	78
Output width	10	10	11	11	12	12	14	13	15	14	14	12	12	12
Hidden fraction	4	4	4	4	4	4	4	4	4	1.5	1.5	1.5	2.5	3
RESULTS														
Convergence?	Y	Y	N	Y	N	Y	Y	N	Y	Y	Y	N	N	Y
Monotonic?	N	N	N	N	N	Y	Y	N	N	N	N	N	N	N
# of trials (epochs)	887	1294	1160	896	4232	1047	495	3874	611	855	659	2207	2078	1662

Table 1: Summary of Fourteen Prototypical Experiments

word than as two words (Hypothesis 3), though most networks were able to perform adequately when each word was represented separately. The input buffer width had a significant effect on network performance (Hypothesis 4), with two word input experiments converging more consistently and faster than one word input experiments. Hypothesis 5 concerns the nature of the hidden unit vectors following training, and whether or not they constitute a "representation" of linguistic knowledge. This will be addressed in the next section.

5. Network Analyses

Principal components analysis and contribution analysis, a variation suggested by Sanger [1989] specifically for evaluating feed forward networks, were used to analyze the hidden unit vectors. The goal of this analysis was to determine if the hidden unit layer acquired a "representation" of the linguistic knowledge as it learned enough about the structure of the presented stories to predict their outcomes (i.e., the direct object of the context-dependent verbs).

Principal components analysis (PCA) consists of several steps, explained briefly in Fukunaga [1972], aimed at determining a coordinate system for a collection of vectors that maximally separates them, i.e., that organizes them into "components". In a feed forward network with hidden units, these components can be viewed as an encoding of the distributed information acquired by the network in training and used by the network to produce desired outputs for particular inputs. The steps of PCA are as follows:

(1) Compute the hidden unit vector corresponding to each input vector;

(2) Compute the covariance matrix of this array of hidden unit vectors;

(3) Determine the eigenvectors of the covariance matrix; these vectors constitute the new coordinate system;

(4) Sort the eigenvectors by their corresponding eigenvalues;

(5) Translate each hidden unit vector into the new coordinate system.

Contribution analysis simply requires that the hidden unit activations computed in step (1) be adjusted by selected weights between the hidden layer and the output layer of the network. The numerical analysis text by Press et al [1989] includes several of the algorithms required to perform eigenvector computation.

Analyses were performed on the hidden unit layer from Experiment 9 (see Table 1): This experiment involved the collection of the four stories shown in Sentences 2a-d, in which the verb "take out" takes four direct objects, which are primed by the semantic context in a previous sentence, and in which the main actor is either male or female. Traditional principal components analysis was performed, as was a contribution analysis focussing on the distributed hidden unit responsibilities toward particular output words.

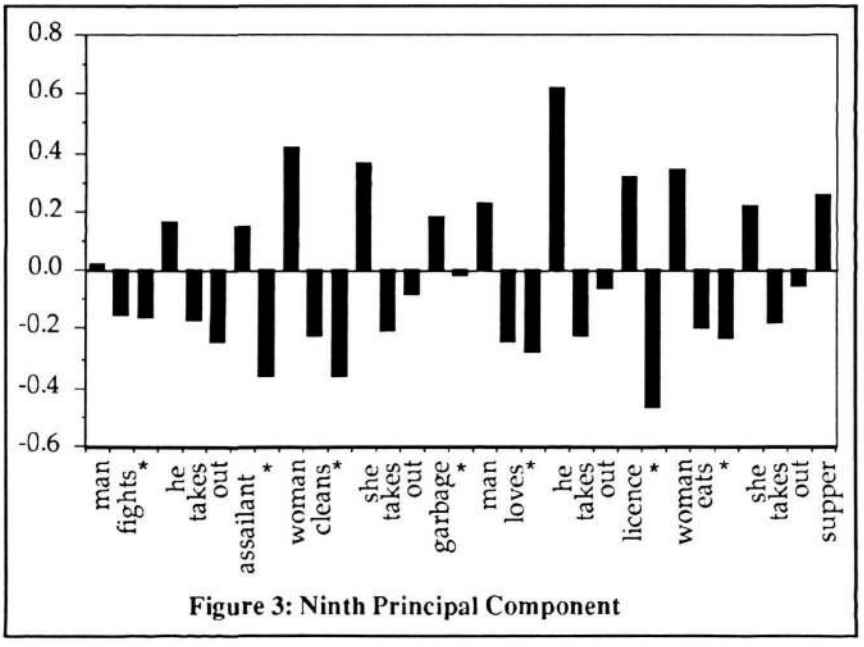


Figure 3: Ninth Principal Component

Three of the most interesting components discovered in this analysis derive from the hidden unit contributions to the output word "garbage", which is one of the primed objects of "take out" that the network learns to predict based on context. These are the components illustrated in the figures. Note that a constant has been added to the raw values to improve the graphic presentation.

Figure 3 consists of a bar

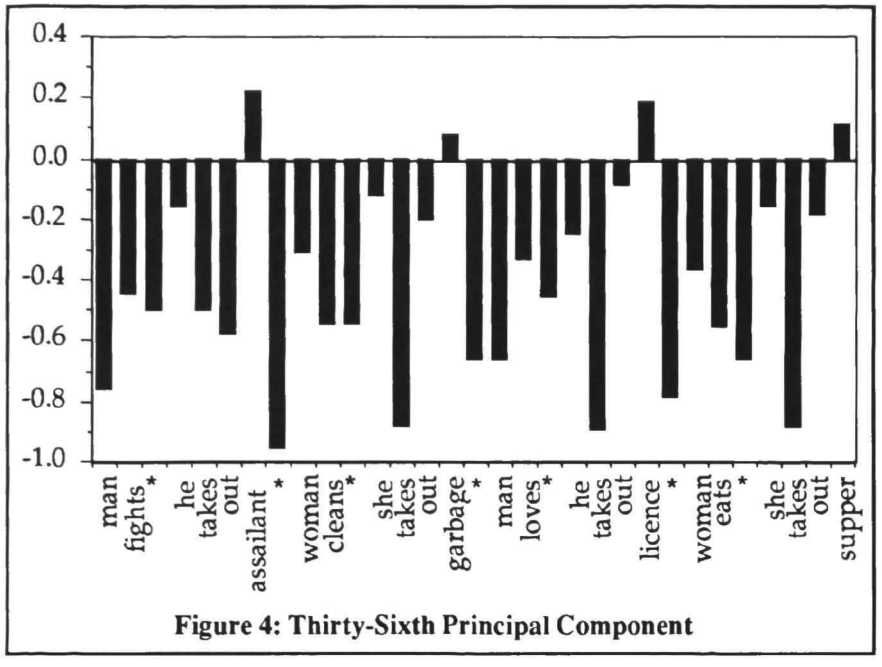


Figure 4: Thirty-Sixth Principal Component

graph illustrating the contribution of the ninth principal component to the decision task of the network. This component discriminates between nouns and other words (i.e., verbs and the periods at the end of sentences). Many of the principal component vectors discriminate among words and word types, and suggest learned linguistic representations (appropriate to the simple linguistic task performed). Other principal components appear not to represent typical linguistic concepts, representing instead heuristically useful information for performing the requested task. In the analyses performed for Experiment 9, for example, one principal component vector seemed to represent a category of pronouns and objects, another represented the words "woman" and all occurrences of the word "take" except the first one, and another the last word of each sentence.

Figure 4 shows that one of the principal components encodes the direct objects of the ambiguous verb "take out". Note that the gender of the sentence subject is subtly represented in these data in the magnitude of the component. The representation of gender contributes to the network's ability to perform the disambiguation task, and can be seen in a number of the principal components, which themselves appear to encode different information. Figure 5 illustrates this point in a principal component that superficially appears to represent the word "out", a word with no semantic role in the experimental stories used here. Significantly, the value of the principal component is maximal when the word "out" occurs in a sentence with a female subject. The network predicts the next word of the input stream by accumulating contextual information; in this case, a readily predictable word that precedes a highly unpredictable one (context excluded) becomes a carrier of contextual information.

As expected, the knowledge gained by these networks, when presented with particular linguistic samples, pertains directly to those samples. The network thus has an inherently heuristic nature; the generalizations (are they representations?) acquired are useful and/or necessary for performing one particular task. Naturally, results based on experiments involving such small corpora of textual samples cannot necessarily be extrapolated to the entirety of human linguistic knowledge and processing.

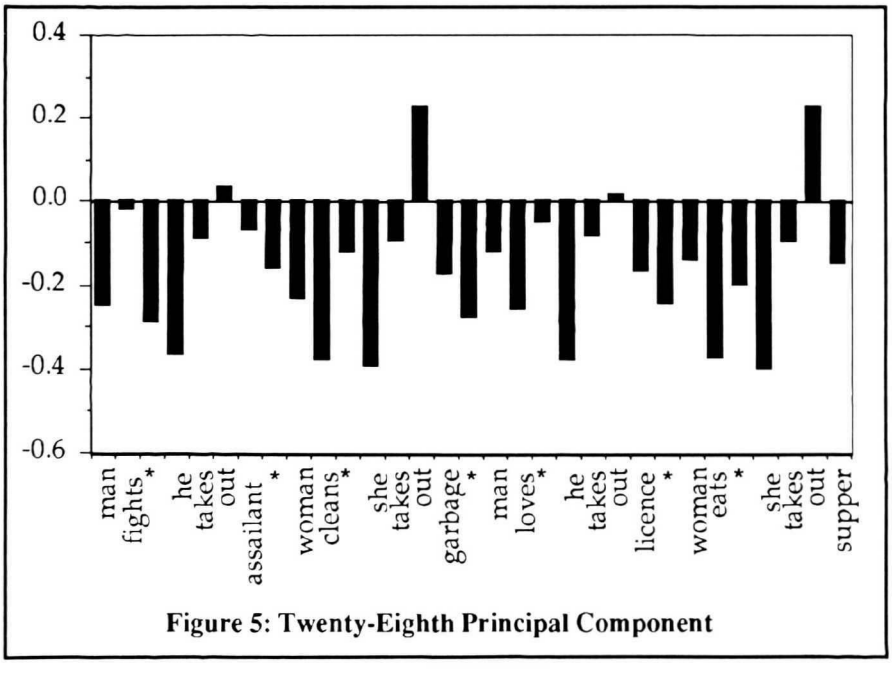


Figure 5: Twenty-Eighth Principal Component

6. Discussion

Much attention has been devoted to the effects of context on human comprehension of sentences and collections of sentences. The relevant context has included local syntactic and semantic features as well as broader elements of textual information. The subject of lexical ambiguity resolution [Small, et al., 1988] has been a productive domain for studies of this type, since understanding the syntactic role and semantics of a word requires knowledge of context at many different levels [Small, 1980].

Linguists have attended to the structural features of sentences and texts that bear on the unambiguous interpretation of subsequent linguistic fragments. Psychologists have employed lexical decision tasks [Tanenhaus, et al., 1979] and auditory evoked potentials [van Petten and Kutas, 1988] to gain information about the temporal sequence of steps performed by the brain to perform word or sentence understanding. While much of this work has been conducted in (presumably) normal users of language, some work has also been done in subjects with language dysfunction, such as Broca's aphasia [Bates and Wulfeck, 1989; Friederici and Kilborn, 1989] or Alzheimer's Disease [Nebes, et al., 1986].

In the current work, a simple recurrent feed forward connectionist network learned to interpret correctly the intended meaning of the words "take out" in context. As noted by Elman [1989], the distributed connectionist approach leads to linguistic performance without explicit rules. Furthermore, the syntactic and semantic structures of language (albeit the very simple examples studied so far) are represented in a distributed non-symbolic form. While in all likelihood, the brain does not employ back propagation learning, it does appear that human learning takes place by weight changes in response to input stimuli (if chemical changes at synapses are viewed as weight changes), and that repetition of stimuli potentiates learning [Lynch, 1986].

7. Conclusions and Future Work

Computational network architectures can learn to perform certain linguistic tasks without any explicitly coded pre-existing linguistic knowledge. In these experiments, simple networks were shown to gain internal linguistic representations sufficient to interpret ambiguous words in context. Furthermore, they were shown to improve performance with (a) shorter distance between contextually important antecedent word and ambiguous word; and (b) increased input buffer size from one word at a time to two words at a time. Both of these processing characteristics have a direct bearing on understanding human performance.

The linear algebraic technique of principal components analysis was used to demonstrate that the network gained a distributed internal representation of various heuristically useful concepts. These concepts include the linguistic notions of "noun" and "direct object", the interesting and useful notion of "the word 'out' in the context of a female agent", and other potentially useful heuristic concepts such as "last word in a sentence" and "period at the end of a contexually important sentence" (i.e., a two word antecedent sentence in one of the simple stories).

Finally, such networks have significant neurological importance. People are subject to a variety of neurological adversities, and the pathophysiology of many are unknown. Computer models of language that can be disrupted to produce deficits analogous to those present in human disease, such as acquired dyslexia [Hinton and Shallice, 1989; Mozer and Behrmann, 1989], may lead to better understanding of these disease processes. In addition to illness, such as stroke and dementia, which produce numerous speaking and understanding (and reading and writing) problems, normal aging also involves changes in linguistic processing. Perhaps a "computational neuropsychology" can shed some light on questions that have been unanswered since Broca [1861].

Acknowledgements

Thanks to the members of the neuroscience community at the University of Pittsburgh who provided helpful comments, advice, and support for the work described here: Audrey Holland, Mark Fitzsimmons, Mac Reinmuth, Gloria Hoffman, and Brad Tanner. Thanks also to Gary Cottrell of UCSD for his help with principal components analysis.

References

Bates, E. and B. Wulfeck, Crosslinguistic Studies of Aphasia, in *The Crosslinguistic Study of Sentence Processing*, MacWhinney and Bates (ed.), Cambridge University Press, Cambridge, 1989.

Blumenfeld, B., A Connectionist Approach to the Recognition of Trends in Time Ordered Medical Parameters, *Symposium on Computer Applications in Medical Care*, Washington, D.C., 1989.

Broca, P. P., Nouvelle Observation d'Aphémie produite par une Lesion de la Partie Postérieure des Deuxième et Troisième Circonvolutions Frontales, *Bullétin de la Societé Anatomique*, 1861, 6: 398-407.

Churchland, P. S. and T. J. Sejnowski, Neural Representation and Neural Computation, Cognitive Neuropsychology Laboratory, The Johns Hopkins University, Technical Report #34, 1987.

Cottrell, G. W. and T. Fu-Sheng, Learning Simple Arithmetic Procedures, *Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, 1989.

Cottrell, G. W. and S. L. Small, A Connectionist Scheme for Modelling Word Sense Disambiguation, *Cognition and Brain Theory*, 1983, 6: 89-120.

Elman, J. L., Representation and Structure in Connectionist Models, Center for Research in Language, University of California, San Diego, Technical Report CRL-TR-8903, 1989.

Fahlman, S. E., An Empirical Study of Learning Speed in Back-Propagation Networks, Computer Science Department, Carnegie Mellon University, Technical Report CMU-CS-88-162, 1988.

Feldman, J. A., Neural Representation and Neural Computation, in *Neural Connections, Mental Computation*, Nadel, Cooper, Culicover, and Harnish (ed.), The MIT Press, Cambridge, 1989.

Friederici, A. D. and K. Kilborn, Temporal Constraints on Language Processing: Syntactic Priming in Broca's Aphasia, *Journal of Cognitive Neuroscience*, 1989, 1: 262-272.

Fukunaga, K., *Introduction to Statistical Pattern Recognition*, Academic Press, New York, 1972.

Hinton, G. E. and T. Shallice, Lesioning a Connectionist Network: Investigations of Acquired Dyslexia, Department of Computer Science, University of Toronto, Technical Report CRG-TR-89-3, 1989.

Lynch, G., *Synapses, Circuits, and the Beginnings of Memory*, The MIT Press, Cambridge, 1986.

Mozer, M. C. and M. Behrmann, On the Interaction of Selective Attention and Lexical Knowledge: A Connectionist Account of Neglect Dyslexia, Department of Computer Science, University of Colorado at Boulder, Technical Report CU-CS-441-89, 1989.

Nebes, R. D., F. Boller and A. Holland, Use of Semantic Context by Patients with Alzheimer's Disease, *Psychology and Aging*, 1986, 1: 261-269.

Press, W. H., B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, *Numerical Recipes in Pascal: The Art of Scientific Computing*, Cambridge University Press, Cambridge, 1989.

Rumelhart, D. E., G. E. Hinton and R. J. Williams, Learning Internal Representations by Error Propagation, Institute for Cognitive Science, University of California, San Diego, Technical Report ICS-8506, 1985.

Sanger, D., Contribution Analysis: A Technique for Assigning Responsibilities to Hidden Units in Connectionist Networks, Department of Computer Science, University of Colorado at Boulder, Technical Report CU-CS-435-89, 1989.

Small, S. L., Word Expert Parsing: A Theory of Distributed Word-Based Natural Language Understanding, Ph.D. Thesis, Department of Computer Science, University of Maryland, 1980.

Small, S. L., G. W. Cottrell and M. K. Tanenhaus (ed.), *Lexical Ambiguity Resolution: Perspectives from Psycholinguistics, Neuropsychology, and Artificial Intelligence*, Morgan Kaufmann Publishers, Inc., San Mateo, California, 1988.

Small, S. L. and C. J. Rieger III, *Parsing and Comprehending with Word Experts: A Theory and Its Realization, in Strategies for Natural Language Processing*, Lenhert and Ringle (ed.), Lawrence Erlbaum Associates, Hillsdale, N.J., 1982.

Tanenhaus, M. K., J. M. Leiman and M. S. Seidenberg, Evidence for Multiple Stages in the Processing of Ambiguous Words in Syntactic Contexts, *Journal of Verbal Learning and Verbal Behavior*, 1979, 18: 427-440.

van Petten, C. and M. Kutas, Tracking the Time Course of Meaning Activation, in *Lexical Ambiguity Resolution: Perspectives from Psycholinguistics, Neuropsychology, and Artificial Intelligence*, Small, Cottrell, and Tanenhaus (ed.), Morgan Kaufmann Publishers, Inc., San Mateo, 1988.

Waltz, D. L. and J. B. Pollack, Massively Parallel Parsing: A Strongly Interactive Model of Language Interpretation, *Cognitive Science*, 1985, 9: 51-74.

How to Describe What? Towards a Theory of Modality Utilization

Yigal Arens and Eduard Hovy*
USC/ISI
ARENS@ISI.EDU, HOVY@ISI.EDU

In this paper we outline the first steps of an investigation of the nature of representations of information, an investigation that uses as a starting point the various ways in which people tend to communicate different kinds of information. Our hope is that by identifying the regularities of presentation, in particular by finding out when people decide to switch presentation modalities and what they then tend to do, we will be able to shed light on the nature of the underlying representations and processes of communication between people.

1 Introduction: The Use of Multiple Modalities

In extended discussions of a technical nature, there invariably comes a point when someone reaches for a pen and draws a diagram or figure. Why? When and why does language, which is after all the most powerful means of communication available to humankind, fall short in expressive power? What additional features do other modalities of communication have? How do our cognitive abilities manage all the disparate kinds of information, splitting them apart during communication and allocating them to various modalities, and then integrating them again? Why is translating to another "visualization" not a simple process — for example, why do some problems seem unsolvable when presented in one way (say, in language) and straightforward when presented in another (say, diagrammatically)? What does it mean for some people to be more "visual" than others?

Questions such as these are interesting to the cognitive scientist because they may shed light both on our internal representations and on our manipulations of them, as reflected by the natures of the modalities we have developed to communicate ideas. This is a deep issue: humans *need* multiple media when they communicate. You cannot speak normally if restricted in hand gesture and facial expression. It is a rare nonfiction book that does not contain photographs, illustrations, or charts. Speakers in most workshops would consider themselves severely handicapped if denied use of overhead or slide projectors. And so forth.

The problem of display design has not yet been given a thorough computational analysis elsewhere. There does exist a general theory of graphical presentation, aimed at the human practitioner ([Bretin 83]), as well as computational treatments of certain subclasses of presentations ([Mackinlay 86, Feiner 88]).

While we do not pretend to have a theory to explain the phenomena, we believe that a careful study of the types of modalities people use, and the types of information they typically utilize them for, will single out characteristics of the underlying cognitive representations and shed light on people's communicative processes. With these issues in mind, initiating a study of the characteristics of representation as expressed through communication, we decided to examine first two aspects:

*This author was supported in part by the Rome Air Development Center under RADC contract FQ7619-89-03326-0001.

- communication-related characteristics of information
- modes of human-human and human-computer communication

We decided to take into account modes that are used in interactions with computers as well, in order eventually to test the rules we develop and implement on a computer against the display decisions made by people.

When identifying characteristics salient to the display of information, the vocabulary should:

- describe all features of the information that are salient for presentation purposes,
- describe all features of presentation modalities that can be utilized to convey information,
- be general enough to allow comparisons and specific enough to differentiate between different modalities and information.

We first define some useful terms, and then provide characterizations of media and information. The paper ends with an example.

2 Characterization of Modalities

2.1 Definition of Terms

The following terms are used to describe presentation-related concepts. We take the point of view of the communicator (indicating where the consumer's subjective experience may differ).

1. **Consumer:** A person interpreting a communication.

2. **Modality:** A single mechanism by which to express information. Examples: spoken and written natural language, diagrams, sketches, graphs, tables, pictures.

3. **Exhibit:** A complex exhibit is a collection, or composition, of several simple exhibits. A simple exhibit is what is produced by one invocation of one modality. Examples of simple exhibits are a paragraph of text, a diagram, a computer beep. Simple exhibits involve the placement of one or more **Information Carriers** on a background **Substrate.**

4. **Carried Item:** That piece of information represented by the carrier; the 'denotation' of the carrier.

For purposes of rigor, it is important to note that a substrate is simply one or more information carrier(s) superimposed. This is because the substrate carries information as well[1]. In addition, in many cases the substrate provides an internal system of semantics which may be utilized by the carrier to convey information. Thus, despite its name, not all information is transmitted by the carrier itself alone; its positioning (temporal or spatial) in relation to the substrate may encode information as well. This is discussed further below.

5. **Channel:** An independent dimension of variation of a particular information carrier in a particular substrate. The total number of channels gives the total number of independent pieces of information the carrier can convey. For example, a single mark or icon can convey information by its shape, color, and position and orientation in relation to a background map. The number and nature of the channels depend on the type of the carrier and on the exhibit's substrate.

2.2 Internal Semantic Systems

Some information carriers exhibit an internal structure that can be assigned a 'real-world' denotation, enabling them subsequently to be used as substrates against which other carriers can acquire

[1] Note that from the information consumer's point of view, Carrier and Substrate are subjective terms; two people looking at the same exhibit can interpret its components as carrier and substrate in different ways, depending on what they already know.

information by virtue of being interpreted within the substrate. For example, a map used to describe a region of the world possesses an internal structure — points on it correspond to points in the region it charts. When used as a background for a ship icon, one may indicate the location of the ship in the world by placing its icon in the corresponding location on the map substrate. Examples of such carriers and their internal semantic systems are:

Carrier	Internal Semantic System
Picture	'real-world' spatial location based on picture denotation
NL sentence	'real-world' sentence denotation
Table	categorization according to row and column
Graph	coordinate values on graph axes
Map	'real-world' spatial location based on map denotation
Ordered list	ordinal sequentiality

Other information carriers exhibit no internal structure. Examples: icon, computer beep, and unordered list.

An internal semantic system of the type described is always intrinsic to the item carried.

2.3 Characteristics of Modalities

In addition to the internal semantics listed above, modalities differ in a number of other ways which can be exploited by a presenter to communicate effectively and efficiently. The values of these characteristics for various modalities are shown in Table 1.

Carrier Dimension: Values: *0D, 1D, 2D.* A measure of the number of dimensions usually required to exhibit the information presented by the modality.

Internal Semantic Dimension: Values: *0D, 1D, 2D, >2D, 3D, #D, ∞D.* The number of dimensions present in the internal semantic system of the carrier or substrate.

Temporal Endurance: Values: *permanent, transient.* An indication whether the created exhibit varies during the lifetime of the presentation.

Granularity: Values: *continuous, discrete.* An indication of whether arbitrarily small variations along any dimension of presentation have meaning in the denotation or not.

Medium Type: Values: *aural, visual.* What type of medium is necessary for presenting the created exhibit.

Default Detectability: Values: *low, medlow, medhigh, high.* A default measure of how intrusive to the consumer the exhibit created by the modality will be.

Baggage: Values: *low, high.* A gross measure of the amount of extra information a consumer must process in order to become familiar enough with the substrate to correctly interpret a carrier on it.

2.4 How Carriers Convey Information

As part of an exhibit, a carrier can convey information along one or more **channels**. For example, with an icon carrier, one may convey information by the icon's shape, color, and possibly through its position in relation to a background map. The number and nature of the channels depends on the type of carrier and the substrate.

The semantics of a channel may be *derived* from the carrier's spatial or temporal relation to a substrate which possesses an internal semantic structure; e.g., placement on a map of a carrier representing an object which exists in the charted area. Otherwise we say the channels is *free.*

Generic Modality	Carrier Dimension	Int. Semantic Dim.	Temporal Endurance	Granularity	Medium Type	Default Detectability	Baggage
Beep	0D		transient	N/A	aural	high	
Icon	0D		permanent	N/A	visual	low	
Map	2D	2D	permanent	continuous	visual	low	high
Picture	2D	3D	permanent	continuous	visual	low	high
Table	2D	2D	permanent	discrete	visual	low	high
Form	2D	>2D	permanent	discrete	visual	low	high
Graph	2D	2D	permanent	continuous	visual	low	high
Ordered list	1D	#D	permanent	discrete	visual	low	low
Unordered list	0D	#D	permanent	N/A	visual	low	low
Written sentence	1D	∞D	permanent	discrete	visual	low	low
Spoken sentence	1D	∞D	transient	discrete	aural	medhigh	low
Animated material	2D	3D	transient	continuous	visual	high	high
Music	1D	?	transient	continuous	aural	med	low

Table 1: Modality characteristics.

Among *free* channels we distinguish between those whose interpretation is *independent* of the carried item (e.g., color, if the carrier does not represent an object for which color is relevant); and those whose interpretation is *dependent* on the carried item (e.g., shape, if the carrier represents an object which has some shape).

Most of the carrier channels can be made to vary their presented value in time. Time variation can be seen as an additional channel which provides yet another degree of freedom of presentation to most of the other channels. The most basic variation is the alternation between two states, in other words, a flip-flop, because this guarantees the continued (though intermittent) presentation of the original basic channel value.

3 Characterization of Information and Its Presentation

In this section we develop a vocabulary of presentation-related characteristics of information.

Broadly speaking, as shown in Table 2, three subcases must be considered when choosing a presentation for an item of information: intrinsic properties of the specific item; properties associated with the class to which the item belongs; and properties of the collection of items that will eventually be presented, and of which the current item is a member. These characteristics are explained in the remainder of this section.

Dimensionality: Some single items of information, such as a data base record, can be decomposed as a vector of simple components; others, such as a photograph, have a complex internal structure which is not decomposable. We define the *dimensionality* of the latter as *complex*, and of the former as the dimension of the vector.

Type	Characteristic	Values
Intrinsic Property	Dimensionality	0D, 1D, 2D, >2D, ∞D
	Transience	live, dead
	Urgency	urgent, routine
Class Property	Order	ordered, nominal
	Density	dense, discrete, N/A
Set Property	Volume	singular, little, much

Table 2: Information characteristics by type.

Since all the information must be represented in some fashion, the following must hold (where *simple* dimensionality has a value of 0, *single* the value 1, and so on, and *complex* the value ∞):

The Basic Dimensionality Rule of Presentations

□ *Rule:* Dim(Info) $\leq$ Dim(Carrier) + Free Channels(Carrier) + Internal Semantic Dim(Substrate)

In addition, we have found that different rules apply to information of differing dimensions. With respect to dimensionality, we divide information into four classes as follows:

- *Simple:* Simple atomic items of information, such as an indication of the presence or absence of email.
  - □ *Rule:* As carrier, use a modality with a dimension value of 0D.
  - □ *Rule:* No special restrictions on substrate.
- *Single:* The value of some meter such as the amount of gasoline left. Associated rule is:
  - □ *Rule:* No special restrictions on substrate.
- *Double:* Pairs of information components, such as coordinates (graphs, map locations), or domain-range pairs in relations (automobile $\times$ satisfaction rating, etc.).
  - □ *Rule:* As substrate, use modalities with internal semantic dimension of 2D.
  - □ *Rule:* As substrate, use modalities with discrete granularity (e.g., forms and tables) if information-class of both components is discrete.
  - □ *Rule:* As substrate, use modalities with continuous granularity (e.g., graphs and maps) if information-class of either component is dense.
  - □ *Rule:* As carrier, use a modality with a dimension value of 0D.
- *Multiple:* More complex information structures of higher dimension, such as home addresses. It is assumed that information of this type requires more time to consume (hence the last rule in this group).
  - □ *Rule:* As substrate, use modalities with discrete granularity if information-class of all components is discrete.
  - □ *Rule:* As substrate, use modalities with continuous granularity if the information-class of some component is dense.
  - □ *Rule:* As carrier, use a modality with a dimension value of at least 1D.
  - □ *Rule:* As substrate and carrier, do not use modalities with the temporal endurance value transient.

- *Complex:* Information with internal structure that is not decomposable, such as photographs.
  - □ *Rule:* Check for the existence of specialized modalities for this class of information.

Transience: Transience refers to whether the information to be presented expresses some current (and presumably changing) state or not. Presentations differ according to:

- *Live:* The information presented consists of a single conceptual item of information (that is, one carried item) that varies with time (or in general, along some linear, ordered, dimension), and for which the history of values is not important. Examples are the amount of money owed while pumping gasoline or the load average on a computer. Most appropriate for *live* information is a single exhibit.
  - □ *Rule:* As carrier, use a modality with the temporal endurance characteristic transient if the update rate is comparable to the lifetime of the carrier signal.
  - □ *Rule:* As carrier, use a modality with the temporal endurance characteristic permanent if update rate is much longer.
  - □ *Rule:* As substrate, unless the information is already part of an existing exhibit, use the neutral substrate.
- *Dead:* The other case, in which information does not reflect some current state, or in which it does but the history of values is important. An example is the history of some stock on the stock market; though only the current price may be important to a trader, the history of the stock is of import to the buyer.
  - □ *Rule:* As carrier, use ones that are marked with the value permanent temporal endurance.

Urgency: Some information may be designated *urgent*, requiring presentation in such a way that the consumer's attention is drawn. This characteristic takes the values *urgent* and *routine*:

- *Urgent:* This situation is exemplified in emergencies, whether they be imminent meltdowns or a warning to a person crossing the road in front of a car. Rules of modality allocation are:
  - □ *Rule:* If the information is not yet part of a presentation instance, use a modality whose default detectability has the value high (such as an aural modality) either for the substrate or the carrier.
  - □ *Rule:* If the information is already displayed as part of a presentation instance, use the present modality but switch one or more of its channels from fixed to the corresponding temporally varying state (such as flashing, pulsating, or hopping).
- *Routine:* The normal case.
  - □ *Rule:* Choose a modality with low default detectability and a channel with no temporal variance.

Density: The difference between information that is presented equally well on a graph and a histogram and information that is not well presented on a histogram is a matter of the density of the class to which the information belongs. The former case is *discrete* information; an example is the various types of car made in Japan. The latter is *dense* information; an example is the prices of cars made in Japan.

- *Dense:* A class in which arbitrary small variations along a dimension of interest carry meaning. Information in such a class is best presented by a modality that supports continuous change:
  - □ *Rule:* As substrate, use a modality with granularity characteristic continuous (e.g., graphs, maps, animations).
- *Discrete:* A class in which there exists a lower limit to variations on the dimension of interest. Appropriate modalities are as follows:
  - □ *Rule:* As substrate, use a modality with granularity characteristic discrete (e.g., tables, histograms, lists).

Volume: A batch of information may contain various amounts of information to be presented. If it is a single fact, we call it *singular*; if more than one fact but still little relative to some some task- and user-specific threshold, we call it *little*; and if not, we call it *much*. This distinction is useful because not all modalities are suited to present *much* information.

- *Much:* The relatively permanent modalities such as written text or graphics leave a trace to which the consumer can refer if he or she gets lost doing the task or forgets, while transient modalities such as spoken sentences and beeps do not. Thus the former should be preferred in this case.
  - □ *Rule:* As carrier, do not use a modality the temporal endurance value transient.
  - □ *Rule:* As substrate, do not use a modality the temporal endurance value transient.
- *Little:* There is no need to avoid the more transient modalities when the amount of information to present is *little*.
- *Singular:* A single atomic item of information. A transient modality can be used. However, one should not overwhelm the consumer with irrelevant information. For example, to display information about a single ship, one need not draw a map.
  - □ *Rule:* As substrate, if possible use a modality whose internal semantic system has *low* baggage.

4 An Example

We present three simple tasks in parallel.
Given: the task of presenting Paris (as the destination of a flight, say).
Available information (three separate examples): the coordinates of the city, the name *Paris*, and a photograph of the Eiffel Tower.
Available modalities: maps, spoken and written language, pictures, tables, graphs, ordered lists.

The modality characteristics are listed among those in Table 1. The information characteristics are listed in Table 3.

The allocation algorithm classifies information characteristics with respect to characteristics of modalities, according to the rules outlined in Section 3. The modality with the most desired characteristics is then chosen to form the exhibit.

Handling the coordinates: As given by the rules mentioned in Section3, information with a *dimensionality* value of *double* is best presented in a substrate with a *dimension* value of *2D*. This means that candidate substrates for the exhibit are maps, pictures, tables, and graphs. Since the *volume* is *little*, *transient* modalities are not ruled out. The value *dense* for the characteristic

	Coordinates	Name	Photograph
Information	48N 2E	Paris	Eiffel Tower
Dimensionality	*double*	*single*	*single*
Volume	*little*	*singular*	*singular*
Density	*dense*	*discrete*	*discrete*
Transience	*dead*	*dead*	*dead*
Urgency	*routine*	*routine*	*routine*

Table 3: Example information characteristics.

density rules out tables. The values for *transience* and *urgency* have no further effect. This leaves tables, maps, and graphs as possible modalities. Next, taking into account the rules dealing with the internal semantics of modalities, immediately everything but maps are ruled out (maps' internal semantics denote spatial locations, which matches up with the denotation of the coordinates). If no other information is present, a map modality is selected to display the location of Paris.

Handling the name: The name Paris, being an atomic entity, has the value *single* for the *dimensionality* characteristic. By the appropriate rule (see Section 3), the substrate should be the neutral substrate or natural language and the carrier one with *dimension* of *0D*. Since the *volume* is *singular*, a *transient* modality is not ruled out. None of the other characteristics have any effect, leaving the possibility of communicating the single word `Paris` or of speaking or writing a sentence such as *"The destination is Paris"*.

Handling the photograph: The photograph has a *dimensionality* value *complex*, for which appropriate rules specify modalities with *internal semantic dimension* of *3D*, and with *density* of *dense* (see Section 3) — animation or pictures. Since no other characteristic plays a role, the photograph can simply be presented.

5 Conclusion

We realize full well that this paper does not present an actual cognitive theory of how people represent and communicate information to highlight various characteristics. However, based on the pervasiveness and regularities in the use of multiple modalities in communication, we believe that any adequate cognitive theory will have to include the types of considerations and characteristics we discuss. That is to say, we believe that notions such as *dimensionality, urgency*, and *granularity* have an irrefutable cognitive reality, thanks to their essential role in the cognitive process of interhuman communication. Future refinements of these terms and identifications of others will help to uncover some of the ways in which people represent and manipulate various types of information.

References

[Bretin 83] Bretin, J. *Semiology of Graphics*, trans. by J. Berg. University of Wisconsin Press, 1983.

[Feiner 88] Feiner, S. An Architecture for Knowledge-Based Graphical Interfaces. ACM/SIGCHI workshop on *Architectures for Intelligent Interfaces: Elements and Prototypes*, Monterey, CA, 1988.

[Hovy & Arens 90] Hovy, E. & Y. Arens. When is a Picture Worth a Thousand Words?—Allocation of Modalities in Multimedia Communication. *AAAI Symp. on Human-Computer Interaction*, Stanford U., 1990.

[Mackinlay 86] Mackinlay, J. *Automatic Design of Graphical Presentations*. Ph.D. dissertation, Department of Computer Science, Stanford University, 1986.

Towards a Failure-driven Mechanism for Discourse Planning: a Characterization of Learning Impairments

Ingrid Zukerman
Department of Computer Science
Monash University
Clayton, VICTORIA 3168, AUSTRALIA
ingrid@bruce.cs.monash.oz.au

ABSTRACT

In the process of generating discourse, speakers generate utterances which directly achieve the communicative goal of conveying an information item to a hearer, and they also generate utterances which prevent the disruption of correct beliefs maintained by a hearer or the inception of incorrect beliefs. In this paper, we propose a representation scheme which supports a discourse planning mechanism that exhibits both behaviors. Our representation is based on a characterization of commonly occurring impairments to the knowledge acquisition process in terms of a model of a hearer's beliefs. As a testbed of these ideas, a discourse planner called WISHFUL is being implemented in the domain of high-school algebra.

INTRODUCTION

In the process of generating discourse, speakers generate utterances which directly achieve the communicative goal of conveying an information item to a hearer†, and they also generate utterances which prevent the disruption of correct beliefs or the inception of incorrect beliefs due to inferences performed by a hearer.

Goal-based discourse planning systems constitute a significant trend in the discourse planning effort. In these systems, a communicative goal, such as *KNOW*(*item*), is posted, and then a plan which includes speech acts as actions is formulated to attain this goal. If, according to a model of a hearer's beliefs, a precondition to an action is not satisfied, then it is posted as a subgoal (Appelt 1982, Hovy 1988, Moore and Swartout 1989). In particular, in systems developed by Hovy and by Moore and Swartout, generated discourse plans include rhetorical relations, such as Elaboration and Evidence, from Rhetorical Structure Theory (Mann and Thompson 1987). Goal-based discourse planners model successfully the first aspect of human discourse generation. However, they fail to account for rhetorical devices such as Revisions of previous material [''*In chapter 7, we saw how to factorize expressions* ... ''] and Contradictions to erroneous beliefs or inferences [''Koalas are marsupials, *not bears*''], which address beliefs that are indirectly affected by the discourse. These rhetorical devices are accounted for by discourse generation systems which apply forward reasoning (Zukerman 1990a). Thus, in order to model both types of human discourse generation strategies and to generate a range of rhetorical devices which supports competent discourse, we need to apply both forward and backward reasoning.

In this paper, we propose a uniform representation formalism which supports discourse planning by means of both types of reasoning processes, and we motivate our representation by means of a simple discourse planner which applies both reasoning processes in sequence. Our formalism relies on a characterization of impairments to the knowledge acquisition process, such as Confusion, Lack of Understanding and Loss of Interest, which is based on a model of a hearer's beliefs.

† The terms speaker/writer and hearer/listener/reader are used interchangeably in this paper.

In the following section, we briefly discuss a model of a listener's beliefs capable of representing uncertain beliefs and predicting inferences commonly drawn in a knowledge acquisition setting. Next, we describe a simple discourse planner which applies both forward and backward reasoning. We then present our characterization of impairments, and demonstrate its application in discourse planning.

NETWORK MODEL OF A LISTENER'S BELIEFS

In order to address beliefs presumably entertained by a particular listener, we maintain an epistemological model which represents a listener's beliefs as a result of direct and indirect inferences drawn from presented material (Zukerman and Cheong 1988, Zukerman 1990a).

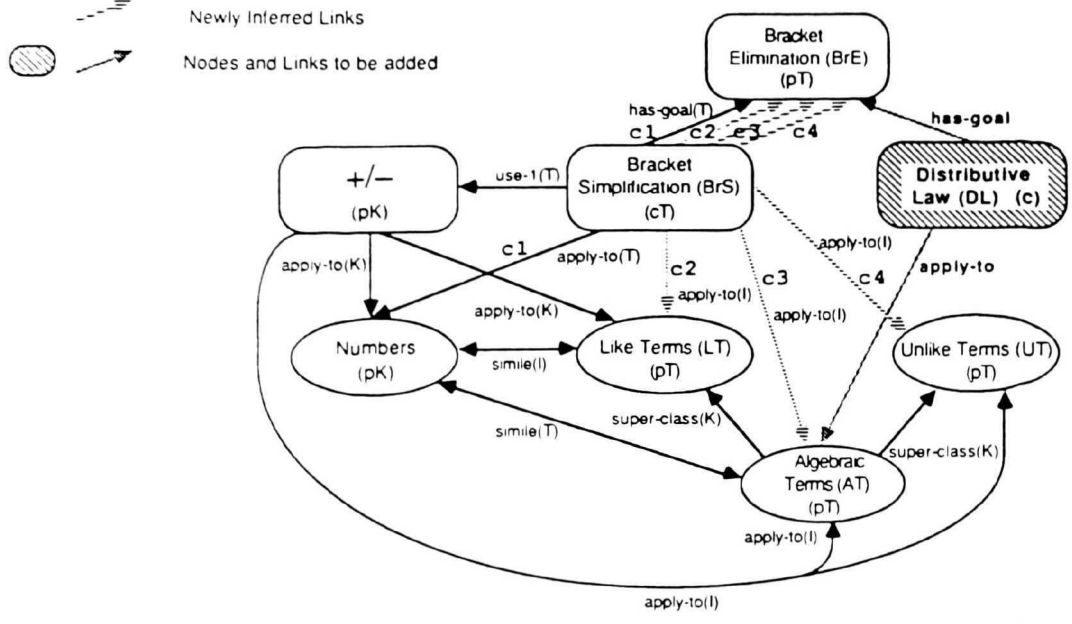


Fig. 1: Network Model of a Listener's Beliefs in High-School Algebra†

We represent a listener's beliefs by means of a network whose nodes contain individual information items and whose links contain the relationships between the nodes (see Figure 1). The links in the network are labeled according to the manner in which they were acquired, i.e., they can either be Inferred, Told or previously Known, where Inferred links are generated by means of generally applicable *Common-sense Inference Rules*. In addition, each link is accompanied by a *Measure of Belief (MB)* between -1 and 1, akin to Certainty Factors (Buchanan and Shortliffe 1985), which represents a user's level of expertise. The information in the network is represented at a level of detail which is consistent with the level of expertise required to learn the subject at hand, i.e., a simple and well-known concept is represented by a p-node (primitive), whereas a relatively new or complex concept is represented by a c-node (complex) which has other nodes as constituents. Like links, nodes may be Inferred, Told or previously Known, and each node has a *Degree of Expertise (DE)* between 0 and 1. The DE of a c-node is a function of the DEs and MBs of its constituent nodes and links, respectively. In this paper, we focus on technical domains, where the transmitted information typically pertains to *procedures*, *objects* and *goals*. Since one procedure will often achieve different goals when applied to different objects, we define a *context* as a triple composed of a procedure, an object to which it is applied, and the goal accomplished by this procedure when applied to this object (labeled c1-c4 in Figure 1).

† In the actual network each link may have a counterpart representing the inverse relationship. However, for clarity of presentation, only links which are relevant to our discussion are shown here.

Our inference mechanism generates plausible inferences from links in the network by means of generally applicable Common-sense Inference Rules which portray reasoning activities such as generalization, specialization and similarity-based inference (see Figure 2). These rules are inspired by rule adaptations commonly performed by students which were studied by Matz (1982), Brown and Van Lehn (1980), Van Lehn (1983) and Sleeman (1984). In order to account for the deductive abilities of a particular type of listener, we annotate each rule with a measure of uncertainty, denoted ρ, which represents a listener's belief in the validity of a conclusion given that the evidence is certain. This measure resembles the rule strength used in ACT* (Anderson 1983).

R1
; If procedure $PROC_a$ initially uses a given set of procedures and these procedures apply to
; disjoint parts of a given object OBJ_m then, with likelihood ρ_1, $PROC_a$ is applicable to OBJ_m
IF [for i=1, · · · ,n ∃ a use-1 link between $PROC_a$ and $PROC_i$ with MB k_{ai}
AND for i=1, · · · ,n ∃ an apply-to link between $PROC_i$ and OBJ_m with MB k_{im}]
THEN (with certainty ρ_1)
Add an apply-to link of type I between $PROC_a$ and OBJ_m with MB $k_{am}=\frac{\rho_1}{n}\sum_{i=1}^{n}k_{ai}k_{im}$

Fig. 2: Sample Inference Rule

THE BASIC MECHANISM

As stated above, in a knowledge acquisition setting, a speaker's communicative goal not only pertains to the acquisition of a particular item of knowledge, but also to inferences a hearer is likely to draw and to other beliefs held by the hearer. Thus, given an *Intended Message (IM)* to be conveyed to a hearer, the following tasks must be performed:

1. Generate *Peripheral RDs*, such as Contradictions and Revisions, which are related to the IM but are not directly instrumental to its acquisition.
2. Generate *Supportive RDs*, such as Causal Explanations, Examples and Descriptions, which are directly instrumental to the attainment of the goal *KNOW*(m), $\forall m \in \{IM, Peripheral\ RDs\}$.
3. Sort the IM and the proposed RDs according to rhetorical considerations.

In general, these tasks should not be performed in strict sequence, but should be interleaved, since they affect each other's results. However, there are conditions under which the sequential execution of these tasks leads to coherent text (Zukerman 1990b). Hence, a two-stage discourse planner which generates Peripheral and Supportive RDs sequentially will suffice in order to illustrate the application of our characterization of learning impairments.

In the first stage, we apply forward reasoning to evaluate the impact of an IM on a listener's beliefs, and generate Peripheral RDs to counteract learning impairments which are likely to take place. To this effect, we temporarily assume that the goal *KNOW(m)* for $m \in \{IM, Peripheral\ RDs\}$ can be attained by merely stating the message in question (this assumption is eliminated in the second step). This stage is executed as follows: initially, a network representing the IM is added to the network representing a listener's beliefs (see Figure 1). Next, a *Recognition* mechanism uses a characterization of learning impairments to anticipate whether the IM is likely to cause an impairment. If this is the case, a *Selection* procedure suggests a preventive Peripheral RD based on this impairment. A *Propagation* mechanism then performs forward reasoning by activating Common-sense Inference Rules to draw inferences from the proposed RD and the IM. The Recognition-Selection-Propagation cycle is repeated with respect to the newly drawn inferences until no impairments remain, i.e., no Peripheral RDs are proposed. This mechanism accounts for the presence of the Contradiction (in italics) in the sentence

"Pandas are bears, *but red pandas are not,*" which prevents Mislearning due to an erroneous inference from the first part of the text. (A detailed description of this procedure appears in [Zukerman 1990a].)

In the second stage, backward reasoning is applied to generate Supportive RDs which fulfill the preconditions for the acquisition of each of the messages proposed in the first stage. Since the fulfillment of the preconditions to discourse actions (i.e., speech acts) is equivalent to the avoidance of learning impairments, a characterization of learning impairments may be used to determine Supportive RDs. This procedure accounts for the generation of a Causal Explanation such as "*they are raccoons*" to support the Contradiction in the above example. In the remainder of this paper we focus on our characterization of learning impairments based on our model of a hearer's beliefs, and demonstrate its use in discourse planning.

CHARACTERIZING KNOWLEDGE ACQUISITION IMPAIRMENTS

We distinguish between three main types of impairments which are responsible for a listener's failure to acquire the beliefs intended by the speaker from the presented information. Our distinction is based on the role of these impairments in the knowledge acquisition process, namely: *Comprehension-related, Affect-related*† and *Inference-related.* Comprehension-related impairments directly inhibit the acquisition of a message, while Affect- and Inference-related impairments inhibit the acquisition of a message through their effect on related beliefs held by a listener. Thus, the recognition and invalidation of possible Comprehension-related impairments is performed in the second stage of the above discourse planning procedure, yielding Supportive RDs; while the recognition and invalidation of the other learning impairments is performed in the first stage, generating Peripheral RDs.

Affect-related Impairments

Affect-related impairments occur when a hearer experiences adverse affective responses due to a conflict between the presented information and his/her existing beliefs. These conflicts inhibit the hearer's acquisition of this information even though s/he may understand it. In a knowledge acquisition setting, two common Affect-related impairments are *Confusion* and *Loss of Interest.*

Confusion occurs when an inference decreases significantly a listener's confidence in a previous belief, i.e., the absolute value of the MB of a link in a network representing a hearer's beliefs is significantly lowered due to the effect of an inference. For instance, upon reading the statement "One cannot always add Algebraic Terms," which yields a negative value for the MB of the link [+/– apply-to AT] in the network in Figure 1, a listener may erroneously infer that one cannot always add Like Algebraic Terms, in direct contradiction with his/her previous belief. The invalidation of Confusion caused by the effect of an incorrect inference on a correct link is performed by a Revision of this link, e.g., "but Like Terms can always be added"; while the invalidation of Confusion due to the effect of a correct inference on an incorrect link is performed by a Contradiction of this link.

Loss of Interest occurs when a listener who is initially motivated to acquire knowledge is presented with an IM s/he considers redundant. In terms of our model, this takes place if there exists a node *B* which *subsumes* a new node *A*, i.e., new distinguishing links incident upon *A* are connected to the same nodes and have MBs of compatible magnitude and sign as the corresponding links incident upon *B*. This situation is illustrated in Figure 1, where we try to add the node ODL, representing distributive law, and the links [DL apply-to AT] and [DL has-goal BrE] to the network representing a listener's beliefs. However, the existence of the erroneous link [BrS apply-to AT] makes the node BrS equivalent to DL, thereby rendering the new procedure redundant and causing Loss of Interest. If Loss of Interest is caused by an incorrect link, it is invalidated by a Contradiction of this link, e.g., "One can not always simplify bracketed algebraic expressions"; whereas if all the links participating in this

† The term *affect* is used in this paper in the sense of *emotions.*

impairment are correct, the generation of a Motivation which adds new links to the message to be transferred is called for. Clearly, Loss of Interest may also be caused by boredom or by lack of understanding. However, in this case, Loss of Interest is a secondary learning impairment which results from other impairments.

Inference-related Impairments

Inference-related impairments take place when a hearer has failed to realize the implications intended by a speaker. That is, inferences which pertain to beliefs that the speaker intends the hearer to hold upon completion of the discourse produce either correct but weak beliefs or incorrect beliefs. These inferences may either affect previously existing beliefs or may be responsible for the inception of new beliefs. Inference-related impairments which are common in a knowledge acquisition setting are *Mislearning, Insufficient Learning* and *Insignificant Change* in a listener's knowledge status. A characterization of these impairments must take into consideration the difference between a listener's level of expertise with respect to a link and a level of expertise considered satisfactory.

Mislearning takes place when an erroneous belief with a relatively high degree of confidence is produced by an incorrect inference drawn by a listener, i.e., the link in question acquires a high MB with a wrong sign. This impairment is invalidated by a Contradiction of the inference.

Insufficient Learning occurs when a correct inference yields a correct belief with a relatively high MB, but which still falls short of a desired MB representative of proficiency. This impairment is invalidated by a Revision of the inference.

Finally, an ***Insignificant Change*** in a listener's knowledge status occurs when an inference produces a rather inconsequential change in a link with an MB representative of insufficient proficiency. The invalidation of this impairment in a link with a relatively high MB is performed by a Revision of the link, if it is correct, and a Contradiction, otherwise. In a link with a low MB, this impairment is considered equivalent to ignorance, and is invalidated accordingly.

The immediate invalidation of Affect-related impairments is essential for the smooth continuation of the knowledge acquisition process, since their persistence diverts a listener's mental resources from the task of acquiring further knowledge. On the other hand, the invalidation of Inference-related impairments with respect to links which are removed from the main focus of the discourse may be postponed if didactic or stylistic constraints prohibit the invalidation of all the recognized impairments.

Comprehension-related Impairments

Different Supportive RDs may accomplish the same function with respect to the comprehension process, e.g., a concept may be created in memory by means of a Description or an Analogy. Further, one Supportive RDs may perform a number of functions, e.g., a Description may be used to create a new concept or to identify a known concept. Therefore, we distinguish between three types of Supportive RDs according to their function rather than their structure, namely *Creative*, *Indicative* and *Explanatory*. Creative RDs are generated to build or reinforce a mental representation of a concept, Indicative RDs are generated to identify an existing concept in memory, and Explanatory RDs are produced to foster belief in a proposition. The Comprehension-related impairments characterized below determine the type of a Supportive RD to be generated.

In order to characterize Comprehension-related impairments, we have found it convenient to separate the comprehension process into three phases: (1) *Access* of the concepts in memory intended by the speaker, (2) *Construction* of a representation in memory of the presented information, and (3) *Acceptance* of the correctness of the presented information. We postulate that in a knowledge acquisition setting, if all these phases are successfully completed, then a message will be understood. In other types of settings, such as a task oriented setting, the third phase is desirable but not essential.

In order to complete the Access phase, the following subgoals must be satisfied: (1) Connection — the hearer must reconcile a referring expression used by a speaker with a node in memory which is intended by the speaker, and (2) Content — the goal *KNOW* must be fulfilled with respect to the intended node. A Content-related impairment may occur in conjunction with a Connection-related impairment, thereby requiring the generation of Supportive RDs which satisfy both subgoals. *Lack of Connection* and *Misunderstanding* are Connection-related impairments, and *Lack of Understanding* and *Insufficient Understanding* are Content-related impairments. These impairments define the preconditions to the attainment of the subgoals of the Access phase, and are characterized in terms of our network model as follows†.

Lack of Connection occurs when one of the following conditions is satisfied: (1) a lexical item used by a speaker to refer to an intended node does not exist in the network which represents a listener's beliefs, i.e., the listener is unfamiliar with the terminology used by the speaker, (2) the lexical item exists in the network, but it is not connected to a concept, (3) it is weakly connected to the intended node (and no other node), or (4) it is connected to the intended node (and no other node), but this node is not primed in the network, i.e., it is outside the listener's *attentional state*‡. The last condition may occur when the discourse diverges both in time and place from the intended node, inhibiting a listener's ability to access it, even if its name has been mentioned. The invalidation of this impairment is performed by means of an Indicative RD, such as the Instantiation in the text "Like Algebraic Terms, e.g., $2x+3x$."

Misunderstanding occurs when a lexical item mentioned by a speaker is connected to a node which is not the intended node. This may be due to a true mis-connection or due to the fact that there is more than one concept with the same name, and the 'wrong' one is primed. A common example of the latter case is a scenario where two people are talking about another person, let's call her Mary, but each participant in the dialogue has a different Mary in mind. Like Lack of Understanding, this impairment is invalidated by means of an Indicative RD, such as "Mary Smith, not Jones."

Lack of Understanding takes place when there does not exist in the network representing a hearer's beliefs a node which corresponds to an intended concept. It entails a connection-related impairment, since a lexical item cannot point to an absent node. This impairment is invalidated by a Creative RD.

Finally, ***Insufficient Understanding*** takes place when there exists a node which corresponds to an intended concept, but the Degree of Expertise associated with this node indicates lack of proficiency. This impairment may occur in conjunction with a connection-related impairment or by itself. It is also invalidated by a Creative RD, but emphasis is placed on addressing missing or erroneous constituents of the node in question, rather than the entire concept. For example, "In completion to square, *you add and subtract* $(b/2a)^2$."

If a lexical item is used by a speaker to refer to a concept, the recognition of an impairment calls for the generation of an *Identification* which associates a proposed Supportive RD with this name, e.g., "A *crook* is a shepherd's staff." This may result in other impairments, such as Confusion or Mislearning, if the intended node is connected (either correctly or incorrectly) to another lexical item, i.e., it is identified with another name, or if the lexical item is connected (either correctly or incorrectly) to another node, i.e., there is more than one node with the same name. Both cases call for the generation of a Revision of the link in question, if it is correct, and a Contradiction, otherwise. In the above example, an impairment may take place if the hearer associates the lexical item *crook* with the concept *criminal*. If a speaker did not use a lexical item, Lack of Connection takes place, calling for the generation of an Indicative RD to enable a listener to access the node in question. In addition, a Creative RD may be required if the listener's expertise with respect to this concept is insufficient.

---

† We assume that the meaning of links, e.g., apply-to and subclass, is understood by a listener, and concentrate on nodes as possible sources of impairments.

‡ The term *attentional state* is due to Grosz and Sidner (1986).

At present, we recognize two preconditions for a ***Construction-related Impairment*** with respect to a given message: (1) the recognition of a Content-related impairment in the Access phase of the comprehension of this message, or (2) a low MB in the links between the nodes in this message or between the constituents of these nodes. The first condition indicates that the hearer is unfamiliar with the concepts themselves, and, hence, is likely to be unfamiliar with the way they relate to each other. It may be invalidated by forcing Instantiations in Creative RDs proposed during the Access phase. The second condition indicates that although the listener may be familiar with the concepts in isolation, s/he is not proficient with respect to the way they relate to each other. This condition may be invalidated by means of Creative RDs with Instantiations with respect to the context at hand.

Finally, we recognize two preconditions whose satisfaction anticipates an ***Acceptance-related Impairment*** with respect to a given message: (1) the application of Common-sense Inference Rules to links in the network which are close to the link representing this message yields an MB for this link which indicates insufficient proficiency, or (2) there exists at least one link which has an MB indicative of an erroneous belief and is related by means of a Common-sense Inference Rule to the link representing the message in question. The first condition stipulates that the combination of the beliefs held by a listener which are related to the belief to be acquired fails to adequately explain the correctness of this belief, whereas the second condition stipulates that it is sufficient to have one belief which undermines the belief to be acquired, in order to anticipate an acceptance failure. Impairments in the Acceptance phase may be invalidated by generating Explanatory RDs, where the links targeted by these RDs are the ones with the lowest MBs. For instance, in the sample network in Figure 1, the Contradiction "Bracket simplification does not always apply to Algebraic Terms" requires a Causal Explanation such as "because you cannot always add Algebraic Terms," if the erroneous link [+/– apply-to AT] has a positive MB. The application of rule R1 in Figure 2 on this link and the link [BrS use-1 +/–] results in the link [BrS apply-to AT], thereby undermining a hearer's belief in the Contradiction.

A WORKED EXAMPLE

In this section, we briefly describe a possible behavior of our discourse planning procedure when teaching a student the distributive law. This situation is represented by the incorporation of the message [DL apply-to AT has-goal BrE], depicted by the shaded node and links in Figure 1, to the rest of the network in Figure 1. The result of this process is summarized in Table 1.

Table 1: RDs Proposed for the Intended Message [DL apply-to AT has-goal BrE]

RD Type	*RD Contents*	*Possible Text*
Revision	**[BrS apply-to LT has-goal BrE]**	We can always eliminate brackets in like terms by applying bracket simplification. E.g., $2(5x+3x) = 2\times 8x = 16x$.
Identification	(BrS, 'bracket simplification')	
Instantiation	(BrS apply-to LT)	
Contradiction	**[BrS ¬apply-to AT]**	However, we cannot do this for all algebraic terms, because we cannot always add algebraic terms.
Causality	(+/– ¬apply-to AT)	
Intended Message	**[DL apply-to AT has-goal BrE]**	In algebra, we can eliminate brackets by applying distributive law: We multiply each term inside the brackets by the term outside the brackets. For example, $2(x+y) = 2x + 2y$.
Identification	(DL, 'distributive law')	
Description	(DL)	
Instantiation	(DL apply-to AT)	

As stated above, in the forward reasoning stage, Loss of Interest is recognized due to the incorrect link [BrS apply-to AT], prompting the generation of a Contradiction of this link. The propagation of inferences from this RD cause Confusion in the correct link [BrS apply-to LT], calling for the generation of a Revision of this link. Since no further impairments are recognized in this stage, we proceed to the backward reasoning stage. During the Access phase, we recognize Lack of Understanding with respect

to the new concept *distributive law*, and Lack of Connection with respect to the concepts *bracket simplification* and *Like Terms*. The first of these impairments is invalidated by means of a Creative RD, such as a Description, and the rest by means of an Indicative RD, say, an Instantiation. Next, during the Construction phase, an impairment may be detected with respect to the application of distributive law to Algebraic Terms, requiring an Instantiation to complement the Description. Finally, in the Acceptance phase, an impairment may be recognized with respect to the Contradiction, calling for a Causal Explanation, as explained above.

CONCLUSION

This paper offers a discourse planning paradigm based on a characterization of impairments to the knowledge acquisition process in terms of a model of a hearer's beliefs. Our characterization provides a parsimonious representation of the preconditions for the fulfillment of a communicative goal, and it supports both backward and forward reasoning in discourse planning. This characterization requires a model of a hearer's beliefs which represents uncertain beliefs and supports the generation of inferences. At present, the forward reasoning process has been implemented with respect to the network in Figure 1, producing Peripheral RDs which are consistent with those appearing in naturally occurring texts.

REFERENCES

Anderson, J.R. (1983), *The Architecture of Cognition*, Harvard University Press, Cambridge, Massachusetts.

Appelt, D.E. (1982), *Planning Natural Language Utterances to Satisfy Multiple Goals*, Technical Note 259, SRI International, March 1982.

Brown, J.S., and Van Lehn, K. (1980), Repair Theory: A Generative Theory of Bugs in Procedural Skills. In *Cognitive Science* 4, pp. 379-426.

Buchanan, B.G. and Shortliffe, E.H. (1985), *Rule-Based Expert Systems — The MYCIN Experiments of the Stanford Heuristic Programming Project*, Addison-Wesley Publishing Company.

Grosz, B.J. and Sidner, C.L. (1986), Attention, Intentions, and the Structure of Discourse. In *Computational Linguistics*, Volume 12, Number 3, July-September 1986, pp. 175-204.

Hovy, E.H. (1988), Planning Coherent Multisentential Text. In *Proceedings of the Twenty-Sixth Annual Meeting of the Association for Computational Linguistics*, State University of New York, Buffalo, New York.

Mann, W.C. and Thompson, S.A. (1987), Rhetorical Structure Theory: A Theory of Text Organization. Technical Report No. ISI/RS-87-190, Information Sciences Institute, Los Angeles, June 1987.

Matz, M. (1982), Towards a Process Model for High School Algebra Errors. In D. Sleeman and J.S. Brown (Eds.), *Intelligent Tutoring Systems*, London: Academic Press, pp. 25-50.

Moore, J.D. and Swartout, W.R. (1989), A Reactive Approach to Explanation. In *IJCAI-11 Proceedings, International Joint Conference on Artificial Intelligence*, pp. 1504-1510.

Sleeman, D. (1984), Mis-Generalization: An Explanation of Observed Mal-rules. In *Proceedings of the Sixth Annual Conference of the Cognitive Science Society*, pp. 51-56.

Van Lehn, K. (1983), Human Procedural Skill Acquisition: Theory, Model and Psychological Validation. In *AAAI-83 Proceedings, American Association for Artificial Intelligence*, pp. 420-423.

Zukerman, I. and Cheong, Y.H. (1988), Impairment Invalidation: A Computational Model for the Generation of Rhetorical Devices. In *Proceedings of the International Computer Science Conference '88: Artificial Intelligence, Theory and Applications*, pp. 294-300.

Zukerman, I. (1990a), A Predictive Approach for the Generation of Rhetorical Devices. To appear in *Computational Intelligence — an International Journal*, Vol. 6, issue 1 or 2.

Zukerman, I. (1990b), Anticipating a Listener's Response in Text Planning. To appear in Golumbic, M.C. (Ed.), *Advances in Artificial Intelligence, Natural Language and Knowledge-based Systems*, Springer-Verlag.

Coherence Relation Reasoning in Persuasive Discourse

Horng Jyh P. Wu and Steven L. Lytinen
Artificial Intelligence Laboratory
The University of Michigan

Abstract

One major element of discourse understanding is to perceive coherence relations between portions of the discourse. Previous computational approaches to coherence relations reasoning have focused only on expository discourse, such as task-oriented dialog or database querying. For these approaches, the main processing concern is the *clarity* of the information that is to be conveyed. However, in a persuasive discourse, such as debates or advertising, the emphasis is on the *adequacy* of presenting the information, not just on clarity. This paper proposes a formalism and a system in which coherence relations corresponding to speech actions such as *clarify*, *make adequate* and *remind* are represented. Furthermore, in relating to human reasoning in general where studies have revealed that implicational and associative reasoning schema are prevalent across various domains, this formalism demonstrates that coherence relation reasoning is similar to this human reasoning, in the sense that coherence relations can be defined by domain independent implicational and associative schema. A prototype system based on this formalism is also demonstrated in this paper in which real world advertisements are processed.

1 Introduction

Some texts hang together better than others. To explain this fact, coherence relations have been proposed, which specify the possible ways in which sentences or other portions of discourse can connect to each other. Using coherence relations, a text's clarity can be evaluated. If an acceptable coherence relation structure can be built, then the text is clear; if not, it is not.

Most of the work in coherence relations has been done on task-oriented dialogues, or on stories. In this sort of discourse, clarity is an essential, and relatively straightforward, feature of the text. For example, in describing a task, the text is coherent if the steps of the task are specified in a well-connected manner. However, coherence is not as simple in persuasive texts, such as debates or advertisements. In these types of discourse, the author's speech actions are constantly affected by precautions such as to be objective or to be polite, etc. Hence, for understanding persuasive discourse, a formalism is needed which can represent those portions of discourse conforming not just to the clarity of the information, but also to the *adequacy* of presenting the information.

An example will illustrate the difference between clarity and persuasive coherence (adopted from Halliday's [HH76], p. 241):

(1.a) Mary is leaving. She was never really happy here.
(1.b) Is Mary leaving? She will be better off somewhere else.

In (1.a), after the second sentence is mentioned, the reason why Mary is leaving becomes clear to the hearer/reader. By being given this reason, the reader shall clarify his picture about the event and understand it better. However, in (1.b), the second sentence is used as a "justification" for the adequacy of the question. It explains why the speaker ask the question. That is, given that Mary shall be happier somewhere else, it is reasonable to ask the question. Similarly, by being given this justification, the hearer may be more willing to cooperate and answer the question as best he can.

In fact, in real life, many discourses can be seen in which an author/speaker will tend to draw support for his requests and beliefs, or try to concede an opinion by acknowledging other facts in order to invite the hearer's cooperation. Such discourses may include debates in an election, arguments in the court, advertisements in magazines or even everyday conversations. Consider, for example, the following two passages:

(2.a) Your oldest son can't help you take care of the other kids. But he can run errands for you.

(2.b) Your invitation is very nice. But I am not available that night.

Although these two utterances would be used in very different situations (one between two mothers, the other between a guest and a host), it is not hard to see the commonality between the two passages. Particularly, they both exhibit a counter-expectation deduced by the following implication-like schema,

The situation is A but it is not so A.

For example, A may stand for BAD in (2.a), since it is "bad" that the oldest kid cannot help take care of the other kids, however it is not so BAD because he can do something else. Similarly, A may stand for GOOD in (2.b), since being invited to a party is "good," but not able to make it is not so "good." To summarize, from a discourse point of view, the first sentence in both (2.a) and (2.b) serves as a concession to hearer's point of view so that they can cooperate and accept the second statement.

In this paper, a formalism is proposed in which coherence relations are treated as individual speech actions connecting portions of discourses. The types of speech actions categorized in this formalisms are *clarify*, *make adequate*, and *remind*. A set of 20 coherence relations originally identified by Mann & Thompson [MT88] are then categorized into the above three classes. For example, for *clarify*, there are coherence relations such as volitional-result and unvolitional-result to indicate the result of events; for *remind*, there are restatement and summary; for *make adequate*, there are evidence, justification, motivation, etc.

A set of implicational semantic relations are summarized from previous psychological studies in our formalism [TvdBS89], [CH85]. The relations included are Goal-oriented (GO), psychological-causation (PSI), physical-causation (PHI), enablement (ENB), obligation (OB), permission (PE), and material-implication (MI). For example, for (1.a) as well as (1.b), the underlying semantic connection between the two sentences is:

If A is unhappy about B then A avoids B.

This is one of the psychological causations between a state and a reaction categorized in Trabasso's [TvdBS89].

Besides implicational semantic relations, there are other underlying relations, which are called associative in our formalism. Following Chaffin and Herman's [CH84] categorization, there are: opposite (OPP), part-whole (PW) and class-inclusion (CI). These semantic relations are believed to underlie coherence relations such as contrast, elaboration and antithesis, which in turn, serve to clarify information. For example, for antithesis,

America rescued Panama. We did not invade it.

There is an opposite (OPP) semantic relation between rescuing and invading, which underlies the relation between the two sentences. The reasons for these categorizations are twofold: first, to capture subcategorizations of coherence relations due to subcategorization in semantic relations; and secondly, as a consequence of the first point, to reveal the hierarchy of coherence relations. The details will be shown in section 3.

2 Defining Coherence Relations (CR's)

In this section, a formalism is introduced which modifies and extends Hovy's system briefly sketched in his paper [Hov88]. The two major "players" in the system are denoted as S-for speaker/author, H-for hearer/reader. Each entity in this formalism has a type and can have associated attributes. Basically, the formalism assumes the representational scheme in a rule-based system such as Soar or Ops5. [1].

The major entities in the formalism are defined in the following.

[1] In the following, the actual representations are abbreviated for clarity of presentation

1. State of affairs, abbreviated as P. State of affairs is the direct semantic representation corresponding to a clause or entities that can be recursively defined as below. For example, "Peter is hungry" is represented as (Hungry Peter).
2. Implication relations, denoted as (IMP P Q), where P and Q are states of affairs. An implication can any one of the 7 types mentioned in section 1. For example, the relation "If A is a policeman then A can investigate people" is represented as (IMP P Q :type Permission), where P and Q denote (Is-a A Policeman) and (Investigate A people), respectively.
3. Associate relations, denoted as (ASSOC P Q), where P and Q are states of affairs. For example, If P is (Rescue America Panama) and Q is (Invade America Panama), then the associated relation between them is Opposite, that is (ASSOC P Q :type opposite).
4. Belief, abbreviated as (BEL X P). A belief has a holder and a content. For examples, the fact that a speaker uses present tense in the sentence "Peter is hungry" indicates his belief that Peter is hungry, denoted as (BEL S (Hungry Peter)).
5. Goal, abbreviated as (GOAL X P). A goal has a holder (X) and the desired state of affairs (P). Goals related to a speaker S are reflected by their speech acts. There are 3 types of major speech acts in this formalism, denoted as SACT (a) Demand actions, denoted as ACT; (b) Inform reference of a entity, denoted as INFORMREF; (c) Inform factual status of an affair, denoted as INFORMYN[2]. For example, a question, "Would you tell me where John is?", is represented as (GOAL S (INFORMREF H S P'), where P' is the proposition (Be-at John ?loc), where ?loc means the location is unknown to the speaker.[3]
6. Mutual belief, denoted as (MB X Y P), where X is the holder of the mutual belief, Y are the other partners who share mutual belief with X, and P, the content of the mutual belief.
7. Mention, denoted as (MEN S P), represents the speech action performed by an author/speaker where S it the speaker and P is the statement being mentioned. Two things should be noted for mentioning. First, the difference between (MEN S P) and (BEL S P) can be demonstrated by the following the sentence,

   Peter might be a student.

   it is translatable as (MEN S P') where P' is (Is-a Peter student) but not as (BEL S P'). Secondly, Mention can takes English words and hence, its implied word order as its arguments). So if speaker says "P because Q" then it can be represented as (MEN S (P because Q)). If the word order is not crucial then a dot is put between the two statements, so (MEN S (P . Q)) means P and Q can be mentioned in any order.
8. Acknowledgment, denoted as (ACK X P), describes an attitude toward a belief or command. In our domain, X is usually the hearer and P is belief or command the speaker wants to implant into the hearer's mind.
9. Coherence relations to make a passage adequate, denoted as (CR S P Q), where CR corresponds to the speech actions Justify, Evidence, motivate, enablement, and concession, were S is the speaker, P and Q are the relevant state of affairs.

Given the above represented entities, coherence relations related to the three major speech actions can be defined as follows[4]

Coherence relations of *make adequate* There are 5 coherence relations included in this category: Evidence, Justification, Motivation, Enablement, and Concession. Let's take Evidence and Justification as examples.

[2]For the latter two types, Cf. Allen's [All83]

[3]Later on, a primed proposition P' of P indicates P' is the declarative counterpart of a question or command expressed by P, or a modal-less propositions of the one with modal

[4]We take the initial coherence relations as those listed in Mann & Thompson's Rhetorical Structure Theory.

```
(Evidence P Q)
If
 (MEM S P) and
 (BEL S Q) and
 (MB (IMP P Q :type all types)) and
[(MEM S (P . Q)) or
 (MEM S (P because Q))]
Then
 (Evidence S Q P) and
 (ACK H (BEL H P'))
```

```
(Justification P Q)
If
  (BEL S P) and
 [(MEM S Q) or
  (GOAL S (SACT H S Q'))]
  (MB (IMP P Q :type all except MI and EMB)) and
 [(MEM S (P . Q)) or
  (MEM S (Q because P)) or
  (MEM S (P so Q))]
Then
  (Justify S P Q) and
 [(ACK H (BEL H P') or
  (ACK H (SACT H S Q'))]
```

The definition for evidence states that if a speaker mentions P, and he also believes in Q (i.e., Q is a declarative sentence), and there is a implicational relation from P to Q, and the speaker mentions "P because Q" or just "P Q," then the speaker uses Q to evidence P so the hearer shall come closer to the belief of P'. An example of the Evidence CR is as follows:

Peter is hungry. (Because) he looks faint.

Since if A is hungry then A will look faint, using the second sentence the reader becomes more convinced of the claim that Peter is hungry. Note that, the above definition will all apply for all seven types of implicational relations. However, in the case of material-implication type, the direction of the implication becomes irrelevant. Hence, the condition (IMP P Q) can also be (IMP Q P :type material-implication). This can be demonstrated by the following examples,

(3.a) Peter might be able to write a poem, because he can write a letter.
(3.b) Peter can write a letter, because he can write a poem.

The second clauses both serve as evidence for the first clauses in (3.a) and (3.b), although the same material-implication is used: If A can write a poem, then A can write a letter.

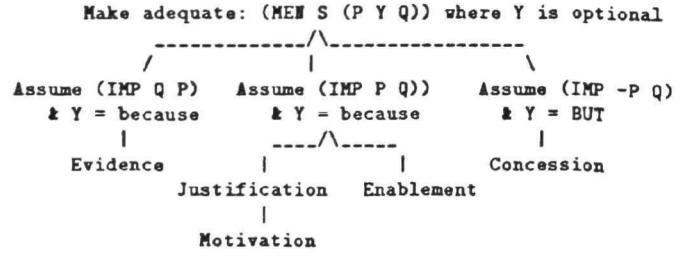
The definition for justification states that, rephrasing the definition backward, if a speaker wants to justify a question, or a command, or a belief described in Q and it is mutually believed that P implies Q, then he can mention P to achieve the justification. An example of a justification coherence relation is as follows:

I am a policeman. Please show me your id.

The speaker justifies his command by indicating he is a policeman, since if A is a policeman, then A is permitted to arrest people (a PE relation). To show the definition works for other implicational relations, for example,

I want to go the London. Can you buy me a ticket by tomorrow?

In this case, the goal-oriented (GO) relation-If A wants to fly to London, then A will buy a ticket to London, together with the mentioning of the motivation, justify the request for the hearer to buy the ticket for the speaker. The other coherence relations together with Evidence and Justification are summarized in the following Taxonomy.



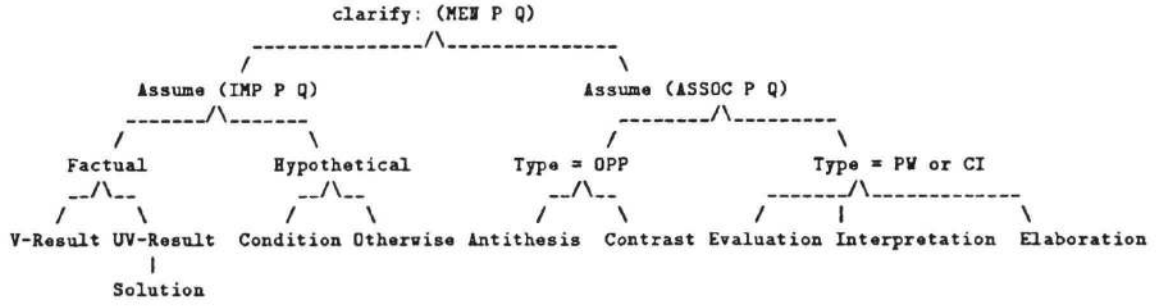
The taxonomy is self-explanatory. The reason why Motivation is under Justification is because Motivation is to Justify a command by resorting to the pleasure or benefit (a PSI type of implicational relation) of the hearer.

Coherence relations of *clarify*. There are ten coherence relations included in this category: Volitional-cause/result, Non-volitional-result/cause, solution, purpose, condition, otherwise, evaluation, interpretation, elaboration, antithesis, and contrast. The definitions for Contrast CR and Antithesis CR are given below:

```
(Contrast P Q)
If
 (MEM S P) and
 (MEM S Q) and
 (MEM S (P Q)) and
 (MB S (ASSOC P Q :type OPP))
Then
 (MEM S (Contrast P Q))
```

```
(Antithesis P Q)
If
  (BEL S -P) and
  (BEL S Q)  and
  (MEM S (P Q))
  (MB S (ASSOC P Q :type OPP))
Then
  (MEM S (Antithesis P Q)) and
  (ACK H (BEL S Q))
```

The definitions indicate that the difference between Antithesis and Concession in of their underlying semantic relations. Antithesis assumes a associative type relation. On the other hand, Concession assumes a implicational type relation. These relations can be summarized in the following taxonomy:



The reason why solution is a especial case of UV-result is because it requires the precedent to be a problem. Evaluation and Interpretation both assume that the situation where the state of affairs is assessed is a part of the entire belief space. The part could be the speaker's individual belief as for Evaluation or another state of affairs as for Interpretation. It is obvious that an Elaboration CR assumes class-inclusion semantical relations.

Coherence relations of *remind* There are two such relations, they are Summary and Restatement. Restatement is a repetition of the exact terms that are mentioned in the previous context, whereas summary is a repetition of some derived terms from the previous context. Their definitions are straightforward. However, it will be shown in the next section that these CR's have an effect on the discourse stack that is not seen in the former two types of CR's.

3 Processing advertisements

The above formalism is currently under development in a system called BUYER (Cf. [LW89]). It takes propositions extracted from real-world advertisements as its input. The final output is the set of coherence relations that are derived from the entire discourse. Following are the control flow of the system and two examples:

The processing control flow.

(0) If end of passage then done, otherwise go to (1), if it is just a sentence or to (2), if it is preceded by a connective.

(1) Processing a sentence:
(1.a) Decide its referential continuity.
(1.a.1) If not continuous then create a new segment and push the old context.
(1.b) Store potential referents in the current segment.
(1.c) Decide the speech acts of the sentences (e.g., it may a BEL, MEN, or GOAL).
(1.d) Abstraction to semantic frames, e.g., to a commodity as follows:
Commodity Kraft-singles:
Brand: Kraft.
Category: Food.
Ingredient: Milk.
Flavor: Tasty.
(1.e) Decide implicational or semantical relations and coherence relations.
(1.f) Go to (0).

(2) Processing a connective between P and Q:
(2.a) Hypothesize the corresponding CR's.
(2.a.1) Start with the most general one if there is no clues for specific ones. For example,
Connective So:
General: Conclusion or Cause.
Specific: Summary or Restatement.
or Justification, etc.
(2.b) Decide the continuity of the discourse.
(2.c) Processing the latter one, Q.
(2.d) Go to (0)

The examples.

The washer ad
S1. It all comes out in the wash.
S2. You can crank-up a GE washer
S3. and get electro-mechanical effect.
S4. Or you can tap-touch a Speed-Queen Marathon
S5. and enjoy fully-programmable laundry.
S6. Match temperature to fabrics.
S7. Recall favorite cycles.
S8. GE or Speed Queen.
S9. The answer is at your figure tips.

The Kraft ad
S1. The older sister said milk is better than oil.
S2. We would never argue with such an authority.
S3. Imitation slices are made from oil and water.
S4. But Kraft singles is made from milk.
S5. Some people already know why Kraft is better than slices.
S6. They are the big brothers and sisters.

We will now describe the processing on the two examples presented above.
The washer ad. After S1 is processed, the referent of "it" is expected to be resolved. The referent is specified as of type PROBLEM, since the undergoer-slot of "comes out" is of PROBLEM type. When S2 is processed, due to referential discontinuity, a new segment is created and the old one pushed. The Coherence relation between S2 and S3 is determined to be volitional-result, since it is one of the CR's that correspond to the connective "and" and there is an underlying implicational relation: If A operates a machine then A gets the effect of the machine. Similar processing happened between S4 and S5, so the Volitional-result CR is decided. At the same time, while processing the connective "or," the relation between S2 and S4 are determined to be the Contrast CR because of the propositional structure corresponding to them is "symmetrical." Furthermore, since the expectation of PROBLEM is stacked, the Contrast CR is interpreted as an exclusive-or (adversative) instead of an inclusive-or (alternative). Thus, the referent of "it" is also resolved. S6 and S7 are evidences to S5 since "If A is a Match-temperature-to-fabric or a Recall-cycles then A is a Programmable-laundry." Meanwhile due to the referential discontinuity, the segment S2-S5 is pushed. Then, S8 is decided as a restatement of the Contrast CR, also the segment S2-S5 is popped because the restatement refers to the terms in the previous segment. R9 is an Evaluation CR, since the judgment references the reader's knowledge. The output can be summarized as:

Elaboration(S1,Contrast(S2,S4)
Volitional-Result(S2,S3)
Contrast(S2,S4)
Volitional-Result(S4,S5)
Evidence(S6,S5)
Evidence(S7,S5)
Restatement(S8,Contrast(S2,S4))
Evaluation(S9,S8)

The Kraft ad. After S1 and S2 are processed, a Justification CR between them is determined, since a implicational relation "If A is an authority then what he says is true" is recognized. S3 and S4 form a new segment because of the referential discontinuity, a Contrast CR between them is also determined, since the propositions corresponding to S3 and S4 have the same predicates but different subjects and objects. Then, the system recognizes when the claim (viz., milk is better than oil) is justified and the contrast related to the claim is given (viz., Kraft singles v.s. imitation slices), the speaker is indicating a exclusive-choice between the contrasted pairs. This causes the system to derive that Kraft singles is better than imitation slices, since milk is better than oil. S5 serves as summary of the above derived facts. S6 is an evaluation of S5, since it references the speaker belief. The output can be summarized as:

Justify(S2,S1)	Summary(S5,S1-S4)
Contrast(S3,S4)	Evaluation(S6,S5)

Top level CR sequence. One interesting observation in advertisement domains is that the top level CR's sequence tends to follow a few fixed patterns. For example, in the above two examples, they both have an assertion (either stated or justified) followed by a contrast, then followed by a Summary and Evaluation. This first part of the sequence resembles a common story plot: background, suspense and resolution. The difference is that in a persuasive discourse, the reader is reminded to reinforce the point. To study these common patterns of persuasion, called debate-plan, is just underway in our investigation.

4 Related work

Being a descriptive theory, Mann&Thompson's [MT88] Rhetorical Structure Theory proposes the initial categorization of coherence relations into presentation and subject-matter, two classes which roughly correspond to our *make adequate* and *clarify* speech actions. However, the definitions given for their Rhetorical relations tend to be too informal for a computational model. Particularly, the theory fails to capitalizes on the fact that the constraints of a Rhetorical relation actually can be conceptualized as implicational and associative semantic relations. Also, the notion of "positive-regard", which was vague in their theory, is represented explicitly in our formalism as the belief and acknowledgment of the reader. Furthermore, their multiple-nuclei Sequence Rhetorical relations, occurring in the top level of the discourse, do not tell us much about the semantic content of the passage. However, as demonstrated in processing advertisements, our system recognizes the sequence to be debate-plans that are commonly seen in persuasive discourse.

Hobbs' theory [Hob], on the other hand, captures of the general semantic relations underlying part of the CR's (those to clarify). However, little emphasis is put on the analysis of the CR's *make adequate* and *remind*.

Hovy's and Mckewon's [Hov88] [McK85] approaches work in the domain of generating answers to data-base queries. The CR's formulated are mostly concerned with clarifying a piece of information. Since their system is for text-generation, it applies knowledge in a back-chaining way. On the other hand, our system applies knowledge in forward-chaining way, and a back-chaining mechanism for understanding a discourse is being investigated for our system.

Allen and Litman's approaches [All83] [Lit86] [Lit87] share the common weakness of previous approaches in missing the CR's *make adequate* and *remind*. Furthermore, for the CR *clarify*, they did not capture the underlying implicational and associative semantical relations, which makes their approaches more domain dependent, especially in their earlier systems.

Cohen's [Coh87] is another computational approach which defines the CR of *make adequate*. However, only the Evidence CR is formulated. The argumentation structure derived from evidence coherence relations thus can not fully cover the debate-plan structure on which most advertisements are based.

5 Future Work

Three types of coherence relations are formulated in this paper: those of *make adequate*, of *clarify* and of *remind*. A prototype system based on this formalism together with other discourse processing mechanisms is shown to process advertisements. However, the formulation, as well as the system, is still rudimentary. Further enhancement can be pursued both in the formalism and the system.

In formalizing discourse reasoning, two goals are set: (1) To refine the formalism's vocabulary, so it make more use of the information deducible from grammatical morphemes for recognizing the implicational semantic relations. (2) To link up the sentence level processing with Unification Grammar based systems and investigate the possibility of even extending the unification mechanism up to the discourse level.

In enhancing the system performance, there are also two goals: (1) To refine the mechanism for deciding referential continuity. Currently the mechanism is simple-minded and works on the advertisements seen so far. A more complicated mechanism may be needed. However, if it turns out there is no need for such mechanism in advertisements, then it may contribute to our understanding of how referential continuity works in debate discourse. (2) To incorporate back-chaining inferencing in implicational relation recognition. According to our observation, advertisements make use of more "immediate" causal relations in consecutive sentences, unlike stories in which causal connections tend to involve several steps. However, there might be cases where hidden connections should exist. In those cases, assumptions should be made and a back-chaining mechanism should be incorporated.

References

[All83] J. Allen. Recognizing intentions from natural language utterances. In M. Brady, editor, *Computational Model of Discourse*, pages 107–166. The MIT Press, 1983.

[CH84] R. Chaffin and D. Herrmann. The similarity and diversity of semantic relations. *Memory and Cognition*, 12:134–141, 1984.

[CH85] P. Cheng and K. Holyoak. Pragmatic reasoning schemas. *Cognitive Psychology*, 17:391–416, 1985.

[Coh87] R. Cohen. Analyzing the structure of argumentative discourse. *Computational Liguistics*, 13:11–23, 1987.

[HH76] M. Halliday and R. Hasan. *Cohesion in English*. Longman, 1976.

[Hob] J. Hobbs. On the coherence and structure of discourse. In Livia Polanyi, editor, *The Structure of Discourse*. Ablex (forth coming).

[Hov88] E. Hovy. Planning coherent multisentential text. In *26th Annual Meeting of the ACL*, pages 163–169, 1988.

[Lit86] D. Litman. Linguistic coherence: A plan-based alternative. In *Proceedings of the 24th Annual Conference of Assoc. for Computational Linguistics*, pages 215–223, June 1986.

[Lit87] D. Litman. A plan recognition model for subdialogues in conversation. *Cognitive Science*, 11:163–200, 1987.

[LW89] S. Lytinen and H. J. Wu. Recognition of coherence relations. In *2nd AAAI Workshop on Plan Recognition*, 1989.

[McK85] K. McKeown. Discourse strategies for generating natural-language text. *Artificial Intelligence*, 27:1–41, 1985.

[MT88] W. Mann and S. Thompson. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8:243–281, 1988.

[TvdBS89] T. Trabasso, P. van den Broek, and S. Y. Suh. Logical necessity and transitivity of causal relations in stories. *Discourse Processes*, 12:1–25, 1989.

Analyzing research papers using citation sentences

Wendy Lehnert, Claire Cardie, and Ellen Riloff
Department of Computer and Information Science
University of Massachusetts
Amherst, MA 01003
E-mail: lehnert@cs.umass.edu

Abstract

By focusing only on the citation sentences in a research document, one can get a good feel for how the paper relates to other research and its overall contribution to the field. The main purpose of a citation is to explicitly link one research paper to another. We present a taxonomy of citation types based upon empirical data and claim that we can recognize these citation types using domain-independent predictive parsing techniques. Finally, an experiment based on a corpus of research papers in the field of machine learning demonstrates that this is a promising new approach for processing expository text.

1 Introduction

One can get a reasonably good understanding of a research paper by merely skimming the sentences that reference other papers. By looking at how the author relates his work to other work in the field, a casual reader can get a good idea of what the paper is all about. As an example, consider the following opening sentence taken from a paper by (Braverman 88):

> The methods of explanation-based learning (EBL) [DeJong & Mooney, 1986] and explanation-based generalization (EBG) [Mitchell, Keller, & Kedar-Cabelli, 1986] involve two conceptual phases: explanation and generalization.

There are two important inferences that can be drawn from the sentence: (1) The paper is pigeonholing itself in the explanation-based learning (EBL) and explanation-based generalization (EBG) paradigms and (2) The distinction between the two conceptual phases is likely to be relevant to the paper. These inferences require general knowledge about research and research papers but no domain-dependent knowledge about EBL or EBG per se. Based on this observation, we contend that research documents can be understood at two distinct levels:

1. **The Semantic Level:** the domain-dependent details of the paper
2. **The Paradigmatic Level:** how the paper relates to other papers in the field

Previous work in understanding expository text has concentrated on summarization at the semantic level. Traditional approaches required extensive domain-dependent knowledge as well as an analysis of the surface structure of the text (see [Britton and Black 85] and [Voss and Bisanz 85] for discussions of these approaches). By focusing on the paradigmatic level of research documents, however, we can ignore the domain-dependent details of a paper and thereby make the task of "understanding" the text a tractable problem. Two main claims follow from this approach:

- Research documents can be understood at the paradigmatic level using a set of *conceptual references*.
- *Conceptual references* can be extracted from research documents using domain-independent predictive parsing techniques.

2 Conceptual References

A *conceptual reference* is a relation between a referencing paper and the referenced paper. Conceptual references represent the reasons behind a citation and tell us why an author references another paper. We have identified two distinct levels of conceptual references: conceptual reference categories and conceptual reference structures. In the following sections, we describe these two levels of conceptual references.

2.1 Conceptual Reference Categories

Conceptual reference categories identify the abstract object that the author is pointing to in the referenced paper. For example, the author may reference another paper to refer to a method, example, or result presented in that paper. Based on the observation that the majority of citation sentences rely on a small set of conceptual reference types, we have created a taxonomy of 18 conceptual reference categories:

1. **System:** a system is described in the referenced paper; e.g. in (Silver 88),
   "Other approaches were considered, including the use of ID3 [Quinlan]..."
2. **Method:** a method is described in the referenced paper; e.g. in (Shavlik 88),
   "Mooney [Mooney 88a] presents an algorithm for generalizing..."
3. **Concept:** a concept is described in the referenced paper; e.g. in (Keller 88),
   "Mostow's original definition of operationality [Mostow 81]..."
4. **Result:** a result is claimed in the referenced paper; e.g. in (Cohen 88),
   "In [Cohen 87] it is shown that PAs are Turing-equivalent..."
5. **Fact:** a fact is stated in the referenced paper; e.g. in (Ellman 88),
   "Standard explanation-based learning (EBL) methods apply only to domains for which a tractable domain theory is available [Mitchell 86]"
6. **Criticism:** a criticism is made in the referenced paper; e.g. in (Hunter 88),
   "A detailed criticism of such purely empirical systems can be found in [Schank 86]"
7. **Example:** an example is used in the referenced paper; e.g. in (Shavlik 88),
   "An example in [Shavlik 88a] shows that ..."
8. **More details:** the referenced paper has more details; e.g. in (Swaminathan 88),
   "For details, the reader is referred to [Swaminathan 88a]"
9. **Attribution:** an item is attributed to the referenced paper; e.g. in (Bylander 88),
   "In [Chandrasekaran 87], three types of explanation are ..."
10. **View:** a view is expressed in the referenced paper; e.g. in (Hirsh 88),
    "Generalization can be viewed as a search problem ([Mitchell 82] [Simon74])..."
11. **Model:** a model is presented in the referenced paper; e.g. in (Mahadevan 88),
    "While there exist formal models for concept learning [Natarajan 87a] ..."
12. **Research:** research is presented in the referenced paper; e.g. in (Clancey 88),
    "Apprenticeship learning research has considered ... [Mitchell 85] [Smith 85]"
13. **Extends:** the referenced paper extends previous work; e.g. in (Cohen 88),
    "[Cohen 88a] has extended the system described in this paper ..."
14. **Application:** the referenced paper presents an application; e.g. in (Bennett 88),
    "For example, Segre has applied EBL to learning robotics tasks in a simplified blocks world [Segre 87b]"

15. **Merge:** the referenced paper merges two techniques; e.g. in (Prieditis 88), *"[this work] is based on combining partial evaluation with other techniques (see Seki and Furukawa [37])"*
16. **Proposal:** the referenced paper proposes an idea; e.g. in (Swaminathan 88), *"[Tadepalli 85] has proposed replacing the original theory with ..."*
17. **Problems:** the referenced paper identifies a problem; e.g. in (Dietterich 88), *"[An important problem] ... is the imperfect theory problem [Mitchell 86]"*
18. **Argument:** the referenced paper argues a position; e.g. in (Ginsberg 88), *"... Kleer gives a similar argument for the ATMS [DeKleer 86]"*

We created this taxonomy of conceptual reference categories based upon an exploratory empirical study of citation sentences. Our corpus contained 372 citation sentences from 40 papers in the Proceedings of the AAAI Spring Symposium Series on Explanation-Based Learning, March 1988.[1] To establish a good set of categories, we set a priori criteria for an acceptable taxonomy: (1) every category must be present in at least 2 sentences from at least 2 different papers (to limit idiosyncracies due to a particular author) and (2) the final set of categories must cover at least 90% of the citation sentences.

First, we arbitrarily selected 208 sentences (the odd-numbered sentences) from our corpus and labelled them by hand with the object being referenced by the citation – its candidate conceptual reference type.[2] Second, we compiled a list of these candidate conceptual reference types and retained only those that occurred in at least 2 sentences from 2 different papers (satisfying the first criterion). Finally, we measured the coverage of this final set and found that it covered 93.7% of the test set. Since this exceeded our 90% threshold (satisfying the second criterion), this set became our 18 conceptual reference categories.

2.2 Conceptual Reference Structures

The conceptual reference categories describe objects being pointed to in the referenced paper. Conceptual reference structures fit on top of these categories to describe the relationship between the referenced object and the current paper. The structures combine conceptual reference categories to explain exactly how a referenced object is being used by the referencing paper. For example, the author might reference a method in another paper to show how his own method *is similar to* the referenced method. There are currently three types of conceptual reference structures: similarity, difference, and flagship references:

- **SIMILARITY:** an object in the current paper is similar to an object in the referenced paper; e.g. in (Ellman 88): *"The general approach ... is similar to methods described in [Keller 87] and [Mostow and Fawcett 87]"*
- **DIFFERENCE:** an object in the current paper differs from an object in the referenced paper; e.g. in (Minton 88): *"Using a different approach, DeJong and Mooney's GENESIS system [DeJong 86b] ..."*
- **FLAGSHIP:** the current paper is pigeonholed via a group of citations of the same type; e.g. in (Prieditis 88): *"See [9,28,29,27,18,35,21] for examples of EBL systems."*

Since these conceptual reference structures fit on top of many conceptual reference categories (e.g. systems, methods, or models can be similar), we can visualize conceptual reference structures as being on a higher plane than conceptual reference categories. A citation sentence may therefore be represented as a conceptual reference category alone (e.g. if a related system is referenced) or as several conceptual reference categories that are embedded in a conceptual reference structure (e.g. if a method is compared to a related method). It is also possible for a sentence to be mapped into

[1] All examples used in the paper are taken from this corpus.

[2] There were actually 223 references because some of the sentences had multiple citations.

more than one conceptual reference if the author references a paper for several reasons. The next section describes how we recognize conceptual reference categories and build structures on top of them.

3 Parsing into Conceptual References

As stated earlier, we claim that understanding the relationships among research papers hinges on the ability to extract conceptual references from a document. Our second claim extends to the parsing mechanisms needed to understand these conceptual references:

1. *The parser relies strictly on predictive parsing techniques.* Predictive parsers are knowledge-based sentence analyzers that create conceptual representations for sentences. They have been used extensively in understanding narrative texts (e.g. [Lehnert 89], [Riesbeck and Schank 76], [Birnbaum and Selfridge 81], [Dyer 83]).

2. *The memory model underlying the parser contains only domain-independent knowledge.* Because we draw our examples from research literature in the field of machine learning, we distinguish between knowledge about research in general (domain-independent knowledge) and knowledge about machine learning (domain-dependent knowledge). In mapping sentences into conceptual references, the parser applies only general research knowledge.

Relationships among research papers are usually made explicit when one research paper references another. For this reason, we can recognize conceptual references without parsing the entire research document and restrict our attention to only those sentences that cite other papers. Citation analysis has been recognized as an important subfield of information retrieval since the early 1960's (e.g., [Garfield 55], [Garfield 64], [Garfield 79]). For the most part, however, researchers in information retrieval have concentrated on statistical analyses of citations to assess the value of an individual paper or the influence of a particular author (see [Salton and McGill 83], [O'Connor 83], and [Hurt 87]). They examine citations only to find the *existence* of explicit links between papers and make no attempt to attach any semantics to the links.

Because our work requires a deeper understanding of the relationships implied when one paper references another, we analyze citations within the context of the sentences in which they occur. These sentences are easily identifiable by a preprocessor that recognizes the peculiar format of citations. Occasionally, there are sentences that contain conceptual references without explicitly citing other papers. There is nothing about our parsing approach that prohibits mapping these sentences into the appropriate conceptual reference types. Identifying these sentences, however, would entail a scan of every sentence in the document. By limiting our attention to explicit citation sentences, we lose little information and only need to parse a small, easily identifiable portion of the text.

The goal of the parser is to represent a citation sentence as an instantiated form of one or more conceptual reference types. Sometimes it is adequate to map a sentence into a simple conceptual reference category. Often, however, a sentence contains more than one conceptual reference or contains one of the more complicated conceptual reference structures. In these cases, the parser returns multiple or embedded representations of the input.

Thus far, we have concentrated on recognizing 10 of the most common reference types described in section 2 — System, Method, Concept, Result, Criticism, Example, More Details, Attribution, Similarity, and Difference references. In the next section, we discuss our approach for parsing citation sentences into one or more of these types. Section 4 walks through a more detailed parse of a sentence from the machine learning corpus.

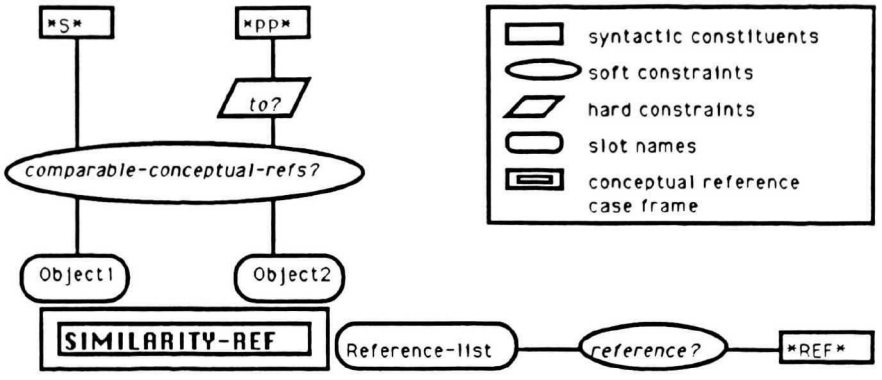


Figure 1: Case Frame Definition for a Similarity Reference

3.1 The CIRCUS Parser

We currently use the semantically-oriented CIRCUS parser [Lehnert 89] for understanding citation sentences. As a conceptual sentence analyzer, CIRCUS represents the meaning of sentences in terms of semantic case frames. Within CIRCUS, a conceptual reference is therefore represented as a case frame structure. Before sentence analysis begins, however, case frame definitions for each conceptual reference type must be hand-coded for the predictive semantics module that performs the slot-filling task. A Similarity reference, for example, has a case frame definition illustrated by Figure 1. Similarity references have three slots: Object1 and Object2 hold the two objects being compared; the Reference-list slot contains a list of citations. In addition, the case frame definition specifies that Object1 will be located in the subject of the sentence, Object2 will be in a prepositional phrase, and the special *REF* syntactic constituent[3] will contain the Reference-list.

Before filling a case frame slot, a syntactic constituent must satisfy the slot's semantic constraints.[4] For a Similarity reference, the prepositional phrase filling Object2 should begin with the preposition "to"[5], the head nouns in the subject and prepositional phrase constituents should be conceptual references that address research at the same level of generality, and the contents of *REF* should be a list of citations.[6] Below we present an example to illustrate how CIRCUS maps a sentence into a Similarity reference using the case frame definition described above.

4 An Example

Consider the following citation sentence:

[3] The sole purpose of the "REF" syntactic buffer is to hold citations.

[4] CIRCUS allows both *hard* and *soft* constraints. A hard slot constraint is a predicate that *must* be satisfied. In contrast, a soft constraint defines a preference for a slot filler rather than a predicate that blocks slot-filling when it is not satisfied.

[5] The Similarity reference case frame in Figure 1 recognizes citation sentences of the form "... {*object1*}...{*"to be" verb*}...{*any synonym for "similar"*}...*to* {*object2*}...".

[6] Because we employ only domain-independent knowledge in mapping citation sentences into conceptual references, the semantic constraints access knowledge about research in general, but do not use any knowledge about the machine learning domain.

Similarity-ref

Object1 : Method-ref
Object2 : Method-ref
Reference-list : ([Keller 87][Mostow and Fawcett 87])

Figure 2: Desired Case Frame

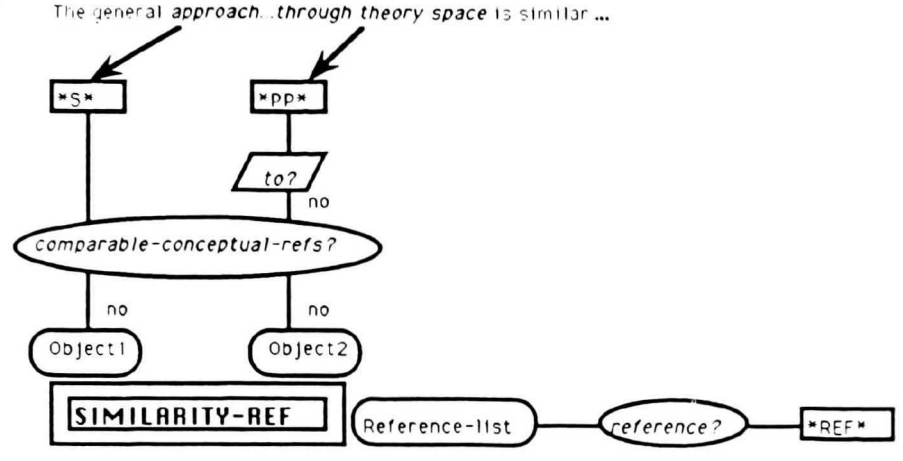


Figure 3: Case Frame Status After the Word "similar"

> "The general approach of using examples to guide search through an approximate theory space is similar to methods described in [Keller 87] and [Mostow and Fawcett 87]" (Ellman 88).

CIRCUS should parse this sentence into the instantiated Similarity reference case frame shown in Figure 2. The parser scans a sentence from left to right, using its stack-oriented control to assign words/phrases to syntactic constituents until it notices a trigger for one of the predefined case frames.[7] Once a case frame is active, CIRCUS' predictive semantics module uses a marker passing algorithm to fill slots in the frame.

In our example, the presence of a "to be" verb followed by the adjective "similar" activates the Similarity reference case frame. In addition, the words "approach" and "methods" are a subset of the phrases that trigger Method references. Figure 3 illustrates the Similarity reference case frame just after CIRCUS has scanned the word "similar". The subject contains a Method reference for "approach" and the most recent prepositional phrase is "through theory space". Although CIRCUS places these constituents in the Object slots (as specified by the case frame definition), the Reference-list slot remains empty. In addition, the semantic constraints associated with Object1 and Object2 have not yet been satisfied. The prepositional phrase filling Object2 should begin with "to" and *both* Object slots should point to conceptual references.

[7] Some Similarity reference triggers are: "Similarly, ...","In the same way, ...", "*X*-like" where *X* can be any noun, "...is the same as...", "...is similar...".

CIRCUS continues scanning the sentence until **all** slots of the active frame(s) are filled without any semantic failures or until it reaches the end of the sentence. After picking up the references, the parser successfully returns the instantiated Similarity reference case frame of Figure 2 because each slot is filled with an object that satisfies the slot's semantic constraints: *REF* contains legitimate references, the preposition in the prepositional phrase constituent is "to", and Object1 and Object2 point to conceptual references that address research at the same level of generality (i.e., both are methods).

5 Evaluation

Our system currently recognizes 10 conceptual reference types — 8 conceptual reference categories (System, Method, Concept, Result, Criticism, Example, More Details, Attribution) and 2 of the higher level conceptual reference structures (Similarity and Difference). It correctly parses 69 sentences from papers in our machine learning corpus and contains over 450 lexicon entries.

To evaluate our progress, we ran an informal experiment. The goal of the experiment was to test the generality of our current set of conceptual reference case frame definitions. We selected two papers from the field of machine learning that were not part of the original corpus[8] and parsed the 28 citation sentences from those papers. We allowed the addition of lexicon entries for any new words occurring in the citation sentences, but did not define any new conceptual reference case frames or case frame triggers.

The system correctly parsed 75% of **all** citation sentences in the two papers. However, two of the sentences (7%) contained conceptual references whose case frames had not been predefined for CIRCUS. (They were not one of the 10 reference types listed above.) Modifying existing conceptual reference frame definitions and adding new triggers allowed 3 more of the remaining sentences to be parsed. With minor modifications to our parser definitions, we could therefore cover 86% of the test sentences. Discounting the 7% covered by undefined case frames, our success rate was then 93%.

6 Conclusion

This paper introduces an original strategy for parsing research documents using conceptual references. The work began in conjunction with the RA document summarization project which aims to summarize scientific research papers in terms of underlying research trends [Swaminathan 90]. RA models domain-independent structural relations in a research field in terms of research schemas and conceptual references. The RA system can use the conceptual references produced by our system to construct a summary of a corpus of research papers.

In closing, we emphasize that our approach to processing expository text is unique in two distinct ways. First, we model domain-independent structural relations between research papers in terms of conceptual references. This is a fundamental departure from other knowledge-based systems that emphasize the semantic content of their domains. Second, we demonstrate that expository text can be processed at the paradigmatic level using strongly predictive techniques. Our results have demonstrated that these two aspects of our system make it a promising new approach for processing expository texts.

[8] In order to remove any bias in the selection of papers, someone outside the parsing development group chose the papers for the experiment, [Keller 87] and [Kedar-Cabelli 87].

7 Acknowledgements

The authors would like to thank Kishore Swaminathan for commenting on earlier drafts of this paper and Stefan Wermter for his work on the project. This research was supported by the Advanced Research Projects Agency of the Department of Defense, monitored by the Office of Naval Research under contract #N00014-87-K-0238, the Office of Naval Research, under a University Research Initiative Grant, Contract #N00014-86-K-0764 and NSF Presidential Young Investigators Award NSFIST-8351863.

References

[Birnbaum and Selfridge 81] Lawrence Birnbaum and Mallory Selfridge, 1981. Conceptual Analysis of Natural Language. In R. Schank and C. Riesbeck, (Eds.), *Inside Computer Understanding*, pp. 318-353. Hillsdale, NJ; Lawrence Erlbaum Associates.

[Britton and Black 85] Bruce Britton and John Black, 1985. Understanding Expository Text: From Structure to Process and World Knowledge. In B. Britton and J. Black, (Eds.), *Understanding Expository Text*, pp. 1-9. Hillsdale, NJ; Lawrence Erlbaum Associates.

[Dyer 83] Michael Dyer, 1983. *In-Depth Understanding: A Computer Model of Integrated Processing for Narrative Comprehension.* Cambridge, MA; MIT Press.

[Garfield 55] E. Garfield, 1955. Citation Indexes for Science. *Science*, 122, pp. 198-111.

[Garfield 64] E. Garfield, 1964. Can Citation Indexing be Automated? In *Proc. of Symposium on Statistical Association Methods for Mechanized Documentation*, Washington, D.C.

[Garfield 79] E. Garfield, 1979. *Citation Indexing — Its Theory and Applications in Science, Technology and Humanities.* New York; John Wiley.

[Hurt 87] C.D. Hurt, 1987. Conceptual Citation Differences in Science, Technology, and Social Sciences Literature. *Information Processing & Management*, Vol. 23, No. 1, pp. 1-6.

[Kedar-Cabelli 87] S. Kedar-Cabelli, 1987. "Formulating Concepts According to Purpose" *Proc. of AAAI*, Seattle, WA.

[Keller 87] R. Keller, 1987. "Defining Operationality for Explanation-Based Learning". *Proc. of AAAI.* Seattle, WA.

[Lehnert 89] Wendy Lehnert, 1989. Symbolic/Subsymbolic Sentence Analysis: Exploiting the Best of Two Worlds. In: Barnden, J. and Pollack, J. (Eds.): *Advances in Connectionist and Neural Computation Theory*, Vol. 1, Ablex Publishers, in press. (Also available as COINS Technical Report 88-99, Department of Computer and Information Science, University of Massachusetts, Amherst, MA.)

[O'Connor 83] John O'Connor, 1983. Biomedical Citing Statements: Computer Recognition and Use to Aid Full-text Retrieval. *Information Processing & Management*, Vol. 19, No. 6, pp. 361-368.

[Riesbeck and Schank 76] Chris Riesbeck and Roger Schank, 1976. Expectation-Based Analysis of Sentences in Context. Research Report No. 78. New Haven, CT; Department of Computer Science, Yale University.

[Salton and McGill 83] Gerard Salton and Michael J. McGill, 1983. *Introduction to Modern Information Retrieval.* New York, NY; McGraw-Hill Book Company.

[Swaminathan 90] Kishore Swaminathan, 1990. "Knowledge Evolution and Memory Organization for Scientific Domains" — a PhD dissertation in preparation. Department of Computer & Information Science. University of Massachusetts, Amherst, MA.

[Voss and Bisanz 85] James Voss and Gay Bisanz, 1985. Knowledge and the Processing of Narrative and Expository Texts. In B. Britton and J. Black, (Eds.), *Understanding Expository Text*, pp. 173-198. Hillsdale, NJ; Lawrence Erlbaum Associates.

Participating in Plan-Oriented Dialogs

Alex Quilici
Artificial Intelligence Laboratory
Computer Science Department
3531 Boelter Hall
University of California
Los Angeles, CA, 90024

Abstract

Participants in plan-oriented dialogs often state beliefs about plan applicability conditions, enablements, and effects. Often, they provide these beliefs as pieces of mostly unstated chains of reasoning that justify their holding various beliefs. Understanding a dialog response requires recognizing which beliefs are being justified and inferring the unstated but necessary beliefs that are part of the justification. And producing a response requires determining which beliefs need to be justified and constructing the reasoning chains that justify holding these beliefs. This paper presents a knowledge-structure approach to these tasks. It shows how participants can use general, commonsense planning heuristics to recognize which reasoning chains are being used, and to construct the reasoning chains that justify their beliefs. Our work differs from other work on understanding dialog responses in that we focus on recognizing justifications for beliefs about a participant's plans and goals, rather than simply recognizing the plans and goals themselves. And our work differs from other work on producing dialog responses in that we rely solely on domain-independent knowledge about planning, rather than on domain- or task-specific heuristics. This approach allows us to recognize and formulate novel belief justifications.

1 Introduction

In discourse processing, two major problems are understanding the underlying connections between successive dialog responses, and producing coherent, cooperative dialog responses. In many plan-oriented dialogs, responses supply pieces of reasoning chains justifying beliefs about why one action should be executed instead of another, about whether a plan has a particular enablement or effect, and so on. In these dialogs, understanding a response involves recognizing its underlying reasoning chain and the beliefs it justifies. And, in these dialogs, producing a response involves formulating an appropriate reasoning chain for a belief in need of justification.

As an example, consider the following dialog fragment.

(1) X: The AI lab members should clean the lab.
(2) Y: But cleaning interferes with research.
(3) X: If we don't clean the lab, who will?
(4) Y: We should pay someone to clean the lab. That way we can do our research.
(5) X: Where do we get the money?
(6) Y: From the fund used to pay salaries.
(7) X: But then we can't hire as many RAs.
(8) Y: I'd rather have a clean office.
(9) X: Then why not clean it up yourself? I'd rather pay an RA than a janitor.

In each utterance X and Y present one or more beliefs, usually as part of a short chain of reasoning justifying or contradicting a belief appearing earlier in the dialog. In (1), X starts by stating a belief that lab members should clean the lab. In (2), Y responds with a belief that lab members cleaning interferes with their doing research. This belief justifies Y's unstated belief that the lab members shouldn't clean the lab. Y's underlying reasoning is that they shouldn't clean the lab because it interferes with their preferred plan of doing research. In (3), X justifies his original belief with the belief that the lab members are the only ones who can clean the lab. X's underlying reasoning is that they should clean it because there's no other way to achieve their goal of keeping it clean. And, in (4), Y states a belief that the lab members should pay someone to clean the lab, and justifies it with the belief that paying someone doesn't interfere with doing research. Y's underlying reasoning is that they should pay someone because doing so doesn't interfere with their goal of doing research, unlike their cleaning the lab themselves. The remainder of the dialog follows the same pattern. In each response, X and Y must recognize which beliefs the other is justifying and the unstated reasoning underlying that justification. And X and Y must also determine which of their beliefs need to be justified and select an appropriate set of justifying beliefs. But how can they accomplish these tasks?

This paper presents a knowledge-structure approach to participating in dialogs involving belief justifications. We show how a dialog participant can use general, common-sense planning heuristics to infer the relationship between stated beliefs and beliefs discussed earlier in the dialog, as well as to construct justifications for their beliefs.

2 Representing Dialog Responses

What types of plan-oriented beliefs appear in the responses in our example dialog? And how can these beliefs be represented?

In our example dialog there are beliefs about whether or not a particular plan should be executed, whether or not a particular plan is the best way to achieve a goal, whether or not one action is more desirable than another, whether or not two actions somehow conflict, whether or not an actor has a particular goal, and whether or not a plan has a particular enablement or effect. We represent these beliefs using *belief*(A,R), where A is the actor (either X or Y) and R is one of the planning relationships shown below in Table 1 (or their negations, which aren't shown, but are denoted as *not-*R). Here, A denotes an actor, S denotes a state (a description of properties of objects), P denotes a plan (a sequence of operators that, when executed, effects a state change), E denotes an event (an actor's execution of a particular plan), and F denotes a filler that can be either a state or an event.

Table 1: Planning relationships in our example dialog.

Relationship	*Semantics*
do(A,P)	A carries out the steps in P
causes(F,F)	F has F' as one of its effects
enables(F',F)	F is necessary for F' to occur
interferes(F,F')	F cannot occur if F does
hasgoal(A,F)	F is a goal of actor A
prefers(A,F,F')	F is more desirable than F'
applies(E,F)	E should occur to achieve F
should-do(A,E)	A should execute E

Essentially, this representation combines elements of other systems that process utterances involving plan-oriented beliefs [4, 5, 9, 15] and extends their representations to include beliefs about planning choices and preferences. More details on this representation can be found in [11], along with examples of its use to represent the plan-oriented beliefs in user/advisor dialogs. Here, however, we illustrate our representation for plan-oriented beliefs by using it to represent the first few responses in our example dialog.

In (1), X's response contains the single belief that the lab members should clean the lab.

should-do(labbies,clean)

In (2), Y states the belief that the lab members cleaning interferes with their doing research. This belief is part of a short reasoning chain that justifies Y's unstated belief that lab members shouldn't clean the lab (a belief that directly contradicts X's originally stated belief).

not-should-do(labbies,clean)
↑ *justifies*
interferes(do(labbies,clean),do(labbies,research)
prefers(labbies,do(labbies,research),do(labbies,clean))

Y's other unstated belief is that the lab members prefer doing research to cleaning. Why can X safely infer that Y holds this belief? Because if Y doesn't hold it, Y's belief that cleaning the lab interferes with doing research isn't a reasonable justification for the lab members not cleaning the lab. To justify not doing an action, that action must conflict with a more desirable action, not one that's less desirable.

In (3), X's stated belief is that no one other than the lab members can clean the lab. This belief is at the bottom of a lengthier reasoning chain.

should-do(labbies,clean)
↑ *justifies*
has-goal(labbies, keep lab clean)
applies(do(labbies, clean), keep lab clean)
↑ *justifies*
causes(do(labbies,clean),keep lab clean)
not-causes(do(other,clean),keep lab clean)

The point of X's reasoning chain is to justify the belief that the lab members should clean the lab. The justification for this belief is a pair of unstated beliefs: that the lab members want a clean lab, and that cleaning the lab is the best way to achieve this goal. This latter belief is, in turn, justified by another pair of beliefs: X's explicitly stated belief that there's no one else to keep the lab clean, and X's unstated belief that the lab members cleaning the lab results in a clean lab.

Finally, in (4), Y provides a pair of beliefs out of a slightly more complex reasoning chain.

applies(do(labbies, clean), keep lab clean)
↑ *justifies*
causes(do(labbies,pay),keep lab clean)
causes(do(labbies,clean),keep lab clean)
preferred(do(labbies,pay),do(labbies,clean)
↑ *justifies*
interferes(do(labbies,clean),do(labbies,research)
not-interferes(do(labbies,pay),do(labbies,research)
has-goal(labbies,do(labbies,research))

This reasoning chain justifies Y's stated belief that the lab members should pay someone to keep the lab clean. This belief is justified by a set of unstated beliefs: that both paying someone and cleaning the lab themselves results in a clean lab, and that the lab members prefer paying someone to cleaning. This latter belief is justified by Y's stated belief that paying someone doesn't interfere with doing research, and by Y's unstated beliefs that

the lab members have a goal of doing research, and that cleaning the lab interferes with this goal.

The remaining dialog responses have a similar representation.

This representation for dialog responses leaves X and Y with two specific tasks. First, they must infer each other's hidden reasoning chains from their stated beliefs. This involves inferring which beliefs are being justified, along with the unstated beliefs that form a necessary part of that justification. And second, they must form the reasoning chains they provide as a response. This involves pulling together a small set of justifying beliefs from their potentially large collection of domain-specific plan-oriented beliefs.

3 Representing Planning Heuristics

Our approach to inferring and forming these reasoning chains rests heavily on two assumptions. The first is that dialog participants use general, common-sense planning heuristics to justify their beliefs. The other is that these heuristics can be represented as collections of abstract planning relationships. Because these heuristics are used to justify beliefs, we call them *justification patterns*.

The first few dialog utterances use several of these heuristics. In (2), for example, Y uses one to justify a belief that the lab members shouldn't clean the lab.

> An actor shouldn't do an action if it interferes with an action the actor prefers.

This heuristic is represented as:

JP:PREVENTS-PREFERRED-ACTION

not-should-do(A,E)
↑ *justifies*
interferes(do(A,E),do(A,E'))
prefers(A,do(A,E'),do(A,E))

Y's response instantiates this JP with the actor as the lab members, the action as cleaning the lab, and the preferred action as doing research.

In (3), X uses a pair of these heuristics. X uses the first to justify the belief that the lab members should clean the lab.

> An actor should do an action if it's the most appropriate action for a goal.

This heuristic is represented as:

JP:BEST-PLAN-FOR-GOAL

should-do(A,E)
↑ *justifies*
has-goal(A,F)
applies(do(A,E),F)

X's response instantiates this JP with the actor as the lab members, the action as cleaning the lab, and the goal as keeping the lab clean. X uses the other heuristic to justify the belief that the lab members cleaning the lab is the best way to keep it clean.

> An actor should execute an action for a goal if no other actor can execute that action.

This heuristic is represented as:

JP:ONLY-ACTION-WITH-EFFECT

applies(do(A,E),F)
↑ *justifies*
causes(do(A,E),F)
not-causes(do(*other*-A,E),F)

X's response instantiates this JP with the actor as the lab members, the action as cleaning the lab, and the effect as keeping the lab clean.

Finally, in (4), Y uses two other heuristics. The first is used to justify his belief that paying someone is the best way to keep the lab clean.

> An actor should execute an action for a goal if it's preferred to an alternative that also achieves the goal.

This heuristic is represented as:

JP:PREFERRED-PLAN-WITH-EFFECT

applies(do(A,E),F)
↑ *justifies*
causes(do(A,E),F)
causes(do(A,E'),F)
prefers(A,do(A,E),do(A,E'))

Y's response instantiates this JP with the actor as the lab members, the preferred action as paying someone, and the goal as keeping the lab clean. Y uses the other heuristic to justify the belief that paying someone is preferred to cleaning the lab themselves.

> An actor prefers one action over another if it doesn't interfere with a goal and the other does.

This heuristic is represented as:

JP:OTHER-PLAN-PREVENTS-GOAL

prefers(A,do(A,E),do(A,E'))
↑ *justifies*
interferes(do(A,E'),F)
not-interferes(do(A,E),F)
has-goal(A,F)

Y's response instantiates this JP with the actor as the lab members, the preferred action as paying someone, the other action as cleaning the lab, and the goal as doing research.

Each type of planning relationship—whether a plan should be executed for an action, whether one action is preferred to another, and so on—is associated with a small set of JPs. Our example dialog contains twelve different justification patterns spanning six different planning relationships.

4 Recognizing Belief Justifications

To understand each other's responses, dialog participants must find a chain of instantiated justification pat-

terns that connect stated beliefs to beliefs that have already appeared in the dialog. Essentially, our approach is to perform a breadth-first search through the space of possible reasoning chains that begin with a stated belief and end with a belief that's directly related to the dialog (matches or negates a belief stated or inferred earlier). The premise is that a participant presents a belief either to justify one of their earlier beliefs or to contradict a belief of another participant.

We illustrate the details of the search algorithm by showing how X understands Y's response in (4). Its input is simply the set of stated beliefs. Here, that means the input is a pair of beliefs: Y's stated beliefs that the lab members should pay someone to clean the lab, and that paying someone doesn't interfere with their doing research.

applies(do(labbies,pay),keep lab clean)
not-interferes(do(labbies,pay),do(labbies,research))

The first step is to determine whether any of these stated beliefs are directly related to the dialog, and, if they are, to mark them as understood. Here, X notices that Y's first belief (that the lab members should pay someone to keep the lab clean) contradicts one of X's earlier beliefs (that the lab members should clean the lab themselves). That means X has now understood Y's first belief, which leaves X to worry only about why Y presented the other belief.

The next step is to form an initial set of potential reasoning chains by finding and instantiating JPs that can contain that belief. One relevant JP is JP:OTHER-PLAN-PREVENTS-GOAL, so X instantiates this JP, resulting in this potential reasoning chain:

prefers(labbies,do(labbies,pay),do(labbies,clean))
↑ *justifies*
interferes(do(labbies,*E'*),do(labbies,research))
not-interferes(do(labbies,pay),do(labbies,research))
has-goal(labbies,do(labbies,research))

There are also other JPs that correspond to Y's stated beliefs, and these JPs are also instantiated and added to the set of potential reasoning chains (but because of space limitations we haven't shown them here).

The next step is to match the beliefs in each of these potential reasoning chains against earlier dialog beliefs. When a match is found, the corresponding reasoning chain is further instantiated with that information. The idea is that the most likely reasoning chains are the ones partially composed of beliefs already in the dialog. In our example, X notices that one belief in the above reasoning chain matches a belief Y stated earlier. This belief, that some action interferes with doing research, corresponds to Y's earlier belief that cleaning interferes with doing research, so X instantiates the reasoning chain with this information.

prefers(*labbies*,do(labbies,pay),do(labbies,clean))
↑ *justifies*
interferes(do(labbies,clean),do(labbies,research))
not-interferes(do(labbies,pay),do(labbies,research))
has-goal(labbies,do(labbies,research))

The final step is to try to recursively connect one of these potential reasoning chains to the dialog. X checks whether any of them justify a belief that directly connects to the dialog. The belief justified by our example reasoning chain (that the lab members prefer paying someone to cleaning the lab themselves) doesn't appear in the dialog. It does, however, fall within another set of JPs. One of these is JP:PREFERRED-PLAN-WITH-EFFECT, which X then instantiates and adds to our example reasoning chain.

applies(do(labbies,pay),*F*)
↑ *justifies*
causes(do(labbies,pay),*F*)
causes(do(labbies,clean),*F*)
prefers(labbies,do(labbies,pay),do(labbies,clean))
↑ *justifies*
interferes(do(labbies,clean),do(labbies,research))
not-interferes(do(labbies,pay),do(labbies,research))
has-goal(labbies,do(labbies,research))

The belief justified by this reasoning chain (that the lab members paying someone is the best way to achieve some goal) corresponds to a belief that appeared earlier in the dialog (Y's stated belief that paying someone is the best way to keep the lab clean), so X instantiates the reasoning chain with this information.

applies(do(labbies,pay),keep lab clean)
↑ *justifies*
causes(do(labbies,pay),keep lab clean)
causes(do(labbies,clean),keep lab clean)
prefers(labbies,do(labbies,pay),do(labbies,clean))
↑ *justifies*
interferes(do(labbies,clean),do(labbies,research))
not-interferes(do(labbies,pay),do(labbies,research))
has-goal(labbies,do(labbies,research))

At this point X has found a reasoning chain that connects Y stated beliefs to the dialog.

Once a participant finds a reasoning chain that connects a stated belief to the dialog, he determines whether the beliefs it contains are shared. This is done by comparing his long-term domains-specific beliefs against the beliefs in the reasoning chain. Doing so fully instantiates the reasoning chain and brings to light any contradictory beliefs, which subsequent responses can then address. In our example, X notices that he fails to share a pair of beliefs in this reasoning chain: the belief that the lab members should pay someone to clean the lab, and the belief that the lab members paying someone results in a clean lab. That latter belief is addressed in X's next response, when he points out that there's no money to pay someone.

In the dialogs we've examined, we've observed no reasoning chain longer than four JPs, so the search terminates unsuccessfully if it can't find a connection to the dialog involving four or fewer JPs.

5 Formulating Belief Justifications

To provide a response, dialog participants must be able to form a reasoning chain justifying one of their beliefs. Our approach once again makes use of the general planning heuristics, this time performing a breadth-first search through the space of possible reasoning chains that might be used to justify a particular belief.

The search for a justification starts with a particular belief to justify. Consider, for example, how Y forms his response in (2). Y wants to justify a belief that contradicts X's belief that the lab members should clean the lab.

not-should-do(labbies,clean)

The first step is to form an initial set of possible justifying reasoning chains by instantiating all the different JPs that can justify the input belief. Here, one of those JPs is JP:PREVENTS-PREFERRED-ACTION, which, when instantiated, results in this reasoning chain.

not-should-do(labbies,clean)
↑ *justifies*
interferes(do(labbies,clean),do(labbies,*E'*))
preferred(labbies,do(labbies,*E'*),do(labbies,clean))

Other JPs for this belief are also instantiated, so there is actually a small set of possible reasoning chains (but due to space limitations we haven't shown any of the others here).

The next step is to instantiate each of these reasoning chains with beliefs already in memory. This task is accomplished by matching beliefs in the reasoning chain with beliefs already mentioned in the dialog and with long-term domain-specific beliefs. Whenever a matching belief is found, the reasoning chain is further instantiated. Our example JP suggests that Y search for a pair of beliefs: that the lab members cleaning interferes with some action, and that this action is preferred to cleaning. Y finds beliefs that cleaning conflicts with doing research, and that research is preferred to cleaning, resulting in the reasoning chain Y presents in (2).

Here, we've assumed that the matching beliefs are easily found. But it's not always possible to successfully instantiate JPs with beliefs already in memory, which means that those beliefs that aren't held in memory must be themselves justified. Consider Y's task in forming his response in (4). There Y is initially trying to justify a belief that there's a better way to keep the lab clean than for the lab members to clean it themselves.

*applies(do(labbies,*E*),keep lab clean)*

One way to justify this belief is JP:PREFERRED-PLAN-WITH-EFFECT, which Y instantiates as:

*applies(do(labbies,*E*),keep lab clean)*
↑ *justifies*
causes(do(labbies,*E*),keep lab clean)
causes(do(labbies,*E'*),keep lab clean)
prefers(labbies,do(labbies,*E*),do(labbies,*E'*))

At this point, Y searches memory for matching beliefs. Y finds one relevant belief from earlier in the dialog: cleaning the lab results in keeping the lab clean. Y instantiates the reasoning chain with that information.

applies(do(labbies,*E*),keep lab clean)
↑ *justifies*
causes(do(labbies,*E*),keep lab clean)
causes(do(labbies,clean),keep lab clean)
prefers(labbies,do(labbies,*E*),do(labbies,clean))

Y also finds a relevant belief in long-term memory: that the lab members paying someone to clean the lab results in keeping the lab clean.

applies(do(labbies,pay),keep lab clean)
↑ *justifies*
causes(do(labbies,pay),keep lab clean)
causes(do(labbies,clean),keep lab clean)
prefers(labbies,do(labbies,pay),do(labbies,clean))

This leaves one belief that Y can't find in memory: that paying someone is preferred to cleaning the lab, so Y must recursively try to justify it. The justification pattern JP:OTHER-PLAN-PREVENTS-GOAL is relevant, so Y instantiates it and adds it to the reasoning chain:

applies(do(labbies,pay),keep lab clean)
↑ *justifies*
causes(do(labbies,pay),keep lab clean)
causes(do(labbies,clean),keep lab clean)
prefers(labbies,do(labbies,pay),do(labbies,clean))
↑ *justifies*
interferes(do(labbies,clean),*F*)
not-interferes(do(labbies,pay),*F*)
has-goal(*labbies*,*F*)

Y now searches memory for matching beliefs. Y finds one relevant dialog belief: that the lab members cleaning the lab interferes with doing research. And we assume Y finds beliefs in memory that paying someone doesn't interfere with research, and that the lab members have that as a goal. Y instantiates the reasoning chain with this information, resulting in the reasoning chain presented in (4).

We actually use a bounded version of this search, limiting potential reasoning chains to a length of four. If, by that point, a dialog participant can't find a fully-grounded reasoning chain, he uses the one containing the fewest ungrounded beliefs. Our rationale is that, in practice, responses rarely provide beliefs in reasoning chains requiring more than three or four JPs to understand.

6 Current Status and Future Work

The model discussed in this paper has been implemented as part of a Prolog program. That program currently consists of two main components, a COMPREHENDER and a CONSTRUCTOR. The input to the COMPREHENDER is a representation for a set of stated participant beliefs. The output is the belief being justified by these stated beliefs, the completed reasoning chain that connects this belief to the dialog, the JPs used to form that chain, and a list of any unshared beliefs in that reasoning chain. The CONSTRUCTOR takes a belief to justify as input and uses the same general planning heuristics to formulate a justification for holding that belief. Its output is the completed reasoning chain that justifies the input belief, along with the JPs used to form that chain.

Currently, we're trying to determine how sufficient our set of justification patterns is for participating in dialogs discussing simple day-to-day planning. We're currently looking at many variants of our AI lab dialog, searching for the presence of other useful justification patterns. And we're using justification patterns for related tasks in other plan-oriented domains, such as recognizing and responding to the misconceptions of novice computer users [10, 11].

We are also working on improving our model's performance. One current problem is that the model assumes that any input belief can be connected to the dialog with a short sequence of justification patterns, and that information from later utterances isn't necessary to complete the connection. But in many dialogs a sequence of responses gradually provide the pieces of a lengthy chain of reasoning, which is impossible to understand until much of the chain is in place. This implies that competing possible justifications should be kept from utterance to utterance, and not thrown away after processing each new utterance, as our model now does. Similarly, when constructing justifications we simply throw away those reasoning chains that couldn't be completely instantiated with beliefs in memory. But it would be better to keep incompletely instantiated reasoning chains around, since the needed beliefs may appear later in the dialog. We're now revising our model to account for this phenomena.

Finally, we're considering various extensions to our model. So far our model suggests one way coherent justifications can be inferred and formed. But our model says nothing about how to compare and evaluate different justifications. And it says nothing about which pieces of the reasoning chains it forms should be presented as a response. Dialog participants, however, are capable of forming a set of coherent justifications for a beliefs and then determining which leads to the best possible response. And dialog participants also rarely present an entire reasoning chain. Instead, they selectively present one or two beliefs. Evaluating reasoning chains and determining which beliefs to present seem to require a detailed model of what beliefs dialog participants hold, beyond the beliefs that are part of the reasoning chains they provide. We're currently working on mechanisms for building this model and then reasoning on it to evaluate the newly-constructed beliefs justifications and to determine which of their beliefs to present.

7 Related Work

Our approach to recognizing and formulating dialog responses owes much to earlier research in story understanding and case-based planning [4, 15, 16, 6, 13]. These systems use high-level, abstract knowledge structures to represent and reason about the underlying theme of the story and to understand their own planning errors. Typically, these knowledge structures consist of a set of abstract planning relationships. Our approach is to use somewhat similar structures for substantially different tasks.

There are several classes of systems working on similar problems. The first consists of systems that attempt to understand the conceptual connections between dialog responses. But these systems [3, 8, 7, 14, 1] primarily focus on recognizing the participant's plan and goals and not on their beliefs about them, such as why one plan should be used instead of another.

The other class of related systems deal explicitly with belief justifications. SPIRIT [9] detects the mistaken beliefs underlying bad plans of users of a computer mail program. It infers the beliefs that led the user to a bad plan, and produces a response that points out these erroneous beliefs. Our task differs in that we're instead trying to recognize the connections between explicitly stated beliefs, and in that we're also concerned with providing responses that justify our beliefs. OpEd [2] recognizes the belief justifications present in economic editorials. But it takes a different approach, relying on linguistic clues and domain-specific reasoning, rather than on more general knowledge about planning. OpEd is also unconcerned with formulating responses that involve belief justifications. Abdul/Ilana [5] tries to participate in arguments that require belief justifications. But it uses a set of task- and domain-specific rules to recognize and understand belief justifications. And it concentrates on selecting from existing justifications for its beliefs, rather than on forming those justifications in the first place. Finally, our own earlier work [10, 11, 12], used a similar, but less general, approach to recognize and respond to the misconceptions of novice computer users. There we relied on significantly more complex justification patterns and provided no method for dealing with chains of reasoning.

8 Conclusions

Previous dialog systems have focused primarily on recognizing a participant's plans and goals. But participants don't simply present their plans and goals. Instead, they present plan-oriented beliefs as part of mostly unstated

reasoning chains that justify their beliefs. To participate in a dialog, it's necessary to recognize which reasoning chains they're using and what beliefs they're trying to justify. And it's necessary to determine which beliefs need further justification and to formulate reasoning chains that justify these beliefs. This paper has presented a knowledge-structure-based approach for accomplishing these tasks.

Our approach is attractive for several reasons. First, it builds a model of many relevant but unstated participant beliefs as a side-effect of trying to relate their utterance to the dialog. It can then reason on this model to help determine where exactly that participant went wrong. Second, it understands belief justifications using the same general, common-sense planning knowledge that it uses to formulate them. That's means the system is capable of understanding any belief justification it constructs. And finally, it shows how novel belief justifications can be understood, so long as they're formed from general planning heuristics known to the participants. That ability is especially important when involved in dialogs between participants that hold differing beliefs, since the participants can't be expected to possess all possible justifications in advance.

References

[1] J.F. Allen and C.R. Perrault, Analyzing intention in utterances. *Artificial Intelligence* 15:143–178, 1980.

[2] S. Alvarado, M. Dyer, and M. Flowers. Editorial comprehension in OpEd through argument units. In *Proceedings of the Sixth National Conference on Artificial Intelligence*. Philadelphia, PA, 1986.

[3] S. Carberry. Modeling the user's plans and goals. *Computational Linguistics*, 14(3):23–37, 1988.

[4] M.G. Dyer. *In-depth understanding: A computer model of narrative comprehension* MIT Press, Cambridge, MA, 1983.

[5] M. Flowers, R. McGuire, and L. Birnbaum. Adversary arguments and the logic of personal attacks. In *Strategies for Natural Language Processing*, Lawrence Erlbaum, Hillsdale, NJ, 1982.

[6] K. Hammond. *Case-Based Planning.* Academic Press, Cambridge, MA, 1988.

[7] H. Kautz and J.F. Allen. Generalized plan recognition. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, 423–427. Philadelphia, PA, 1986.

[8] D.J. Litman and J.F. Allen. A plan recognition model for subdialogues in conversations. *Cognitive Science*, 11:163–208, 1987.

[9] M. Pollack. A model of plan inference that distinguishes between the beliefs of actors and observers. In *Proceedings of 24th meeting of the Association of Computational Linguistics*, 207–214, New York, NY, 1986.

[10] A. Quilici. The Correction Machine: Using justification patterns to advise novice UNIX users. In *Intelligent help systems for UNIX: Case studies in Artificial Intelligence*, R. Wilensky, P. Norvig, and W. Wahlster, editors. Springer Verlag, New York, NY, 1990.

[11] A. Quilici. The Correction Machine: Formulating explanations for user misconceptions. In *Proceedings of the 11th Joint Conference on Artificial Intelligence*, Detroit, MI, 1989.

[12] A. Quilici, M.G. Dyer, and M. Flowers. Recognizing and responding to plan-oriented misconceptions. *Computational Linguistics*, 14(3):38–51, 1988.

[13] R. Schank. *Dynamic Memory.* Cambridge University Press, Cambridge, MA, 1982.

[14] C. L. Sidner. Plan parsing for intended response recognition in discourse. *Computational Intelligence*, 1(1):1–10, 1985.

[15] R. Wilensky. *Planning and Understanding*, Addison Wesley, Reading, MA, 1983.

[16] R. Wilensky. Story Points. In *Strategies for Natural Language Processing.* Lawrence Erlbaum, Hillsdale, NJ, 1982.

Fictional Narrative Comprehension: Structuring the Deictic Center

Erwin M. Segal
Center for Cognitive Science and
Department of Psychology
SUNY, University at Buffalo

Abstract

An analysis of the structure of sentences found in fictional text, and the interpretation that one gives to them has led to the proposal that all fictional text is written from a perspective within the fictional world of the story. In a like manner, readers read the story from a similar perspective. The author "pretends" that he is in the story by locating an image of herself somewhere within the space-time of the story (even at times within characters of the story) and creates the sentences from that vantagepoint. The story and its sentences must contain cues so that readers can use the text to discover the perspectival sources of the sentences. They can then pretend that they are "in" the story and can read it from those perspectives. The perspective from which the sentences are read is called the "Deictic Center." This proposal is associated with ongoing research to implement a cognitive model which reads fictional text according to these principles.

Over the past several years I and my colleagues in the Narrative Comprehension Research Group at the University at Buffalo (SUNY) have been studying narrative text using a theoretical frame that we now call the Deictic Shift Theory (Bruder et al., 1986; Rapaport et al., 1989; Galbraith, 1989) This model assumes that a reader comprehends many of the sentences in a fictional narrative by shifting an image of himself to locations in the story world. These locations, serve as the source points for the interpretation of the sentences of the text. The reader uses specific linguistic devices found in the text to control shifting from one source point to another.

In this paper I use logical argument and linguistic examples to support the Deictic Shift Theory. I start by introducing the deictic nature of tense in conversation and then I attempt show how it has been applied to fictional text. The Deictic Shift Theory is then presented as an attempt to resolve the difficulties identified.

Contextualization. Fillmore (1981) showed that certain text requires a particular contextual frame in order for it to be uttered and understood in a conventional manner. This example he used from Hemingway's "The Killers" is best understood by reading it from a perspective inside Henry's lunchroom.

(1) "The door of Henry's lunchroom opened and two men came in."

Fillmore states:

> It seems to me that the discourse grammarian's most important task is that of characterizing, on the basis of the linguistic material contained in the discourse under examination, the set of worlds in which the discourse could play a role, together with the set of possible worlds compatible with the message content of the discourse.(1981, p. 149)

This analytic process he calls "contextualization." Fillmore shows that certain text is meaningfully uttered only in certain situations. If our goal is the representation of the process of comprehension, we need to incorporate the idea of contextualization in our models; that is, we need to articulate a mechanism which generates and modifies the contextual frame as the sentences are being comprehended.

The Deictic Shift Theory is based on a generalized application of the principle that certain terms are "token reflexive" (Reichenbach, 1947) and thus comprehension is constrained by the situation (Barwise & Perry 1983). Most obviously, certain words such as here, now, today, this, I, you, ago, etc. are referenced as a function of the specific time and place of the utterance rather than simply as a function of their meaning. Obviously deictic terms play an important role in everyday discourse. They play an important role in how we use our language to talk of the objects and events around us.

Tense. Reichenbach (1947) included verb tense in his class of deictic terms. Many sentences are to be understood as applying to a particular event which occurs at a particular place at a particular time. The time of the event is usually identified in relation to the act of utterance rather than by some context independent description. In most discourse, if an event precedes the utterance which describes it, it would be described in a past tense. If the time of the event is simultaneous with the time of its description it would be described in a present tense. If the description is of a predicted or expected event a future tense would be used.

Reichenbach's (1947) explanation of different tenses was in terms of three variable points or ranges on a time line which passes through the time at which the utterance is made. These points are speech time, reference time, and event time, S, E, and R, respectively. Different tenses are used dependent upon the order on the time line of these three points. Consider the event E <John eat>. If his eating took place before R, some other time of significance which also has already taken place, and I am expressing this fact now, S. The sentence uttered would be in the past perfect, (2) John had eaten, and the relevant times could be diagrammed this way

```
(2) --|-----|-----|---> t   past perfect
      E     R     S
```

The horizontal line represents the passage of time, the vertical bars represent the points on the line that the tree events take place. The

bar above the S represents an image of me on the time line, when I am speaking the sentence John has eaten. Reichenbach proposed that these three factors were all involved in all tenses. If the simple past were used, (3) John ate, the relevant reference time is the same as the event time.

```
(3) --|-----|---> t   simple past
      E,R    S
```

The present perfect, (4) John has eaten, differs from the past in that reference time is the same as speech time.

```
(4) --|-----|---> t  present perfect
       E    R,S
```

Other basic tenses, present, (5) John eats, future, (6) John shall eat, and future perfect (7) John shall have eaten, are also defined by the three points on the time line,

```
(5) --|---> t present      (6) --|-----|--> t  future
    S,E,R                         S,R    E

(7) --|-----|-----|--> t  future perfect
       S     R     E
```

For multiple clauses in the same sentence R remains constant. Thus the two clauses in (8) John had eaten when Mary entered share R, the time that Mary entered. Reichenbach's approach has been discussed, clarified, and extended by many researchers (cf. Schopf, 1987, 1989), but with few exceptions, such as Casparis's (1975) argument that tense does not mark temporal relations, his basic premise that tense marks relations on a time line which deictically redounds back to the speaker tends to be accepted.

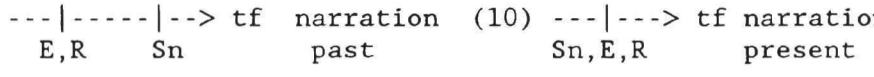
Reichenbach's analysis was on the deictic nature of tense, but he described it entirely in terms of the speaker. Deixis is not solely an issue of the speaker of the sentence, but also of the hearer (reader, addressee, etc.) If one of our interests is that of comprehension, we must consider where and when the hearer or reader is when the communication is being received.

If deictic terms are being used, the hearer must be cognizant of the situation in order to interpret those terms correctly. In the normal unmarked form, and for the above cases, we can assume that the hearer's location on the time-line is simultaneous with the speaker's. Although if the text is written, one might assume on logical grounds that the reader's situational context does not correspond to the writer's, but occurs at a later time and in an unspecified place. How shall deixis be comprehended in such cases? Deictic terms are used quite extensively in written narrative text, particularly fictional text.

Fiction. In English the primary tense of a fictional narrative is usually past, although the present is occasionally used (Casparis, 1975). This is true regardless of the temporal relation to the author or

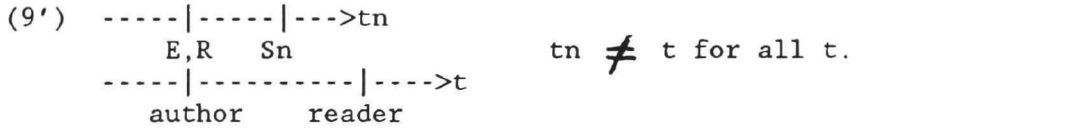
the reader. (It is argued by Buhler (1934) that in fiction the events can never be based on the author's spatial or temporal coordinates.) Past tenses may be used whether the narrative is about some prehistoric events which are purported to have occurred 30,000 years ago, or is science fiction which describes events that may occur 30,000 years in the future. Moreover, the primary tense usually remains the same throughout the text. Regardless of this fact, temporal relationships among the events are maintained and understood. Obviously the time of the author or reader is not directly relevant to the way that tense maintains the temporal interpretation of the textual events. Thus tense in narrative is not token reflexive in the Reichenbachian way. If tense in fiction is deictic at all, it has to be differently contextualized. We need to identify a substitute for the utterance situation in fictional narrative.

One semantic explanation for certain of the narrative uses of tense is through the invention of a fictional speaker or narrator who tells the story. The fictional time line, tf, in this analysis goes through Sn, the narrator's "speech time." If the narrative is in the past tense then the fictional narrator is telling the story at some time after the event that is described. If in the present tense the narrator describes events which are concurrent to his fictional telling. (9) represents John ate as told by a narrator of a fictional story; and (10) represents a narrator asserting John eats in what is usually called the 'historical present' tense.



A fictional narrator is a major component of the poetics of fiction. There are many fictional narratives in which a narrator is explicitly included in the text (cf. Booth, 1983). Most narrative theorists, however, include a fictional narrator even when the text does not explicitly identify one (eg. Genette, 1988; McHale, 1983; Cohn, 1978).

This analysis seems to solve many problems. The hypothesis that a fictional narrator utters the sentences of the text meaningfully contextualizes the sentences in the fiction. The fictional events were witnessed by the fictional narrator. The narrator is "in the fiction" where s/he can ground the deictic references and where s/he may have "witnessed" the story events and thus can epistemologically justify the events described. However, it addresses neither the role of the 'real' author nor that of the 'real' reader. They exist on a time line and in a situation which have no obvious connection with those in the fictional story world.



The ontological problem of what it means to have a story told by a fictional narrator tends not to be explored, although Hamburger (1973) rejects inventing fictional narrators partially on ontological grounds.

In addition to the ontological ones, there are among others, cognitive and linguistic problems with the assumption that a fictional narrator tells the story. Often the syntactic structure of narrative sentences cannot be grammatically tied to a narrator; how are these sentences to be understood? How do readers interpret deictic terms designed to be used in their presence such as here and now? Why is it that some sentences are not syntactically an assertion, or not even complete sentences? How do we comprehend them? Why should a narrator telling a story use such linguistic forms? These are seen to mark expressive elements (Banfield, 1982), and often they do not seem to be expressions of the narrator.

I think that the role of the author in writing fiction is important in understanding its comprehension. One view on authors is that of Searle (1975a) who analyzed fictional text with the focus on understanding the author's speech act in generating the text. He proposed that authors pretend to use standard dialogic speech acts (cf. Searle, 1969, 1975b). For the most part they pretend to assert the propositions in the text, and they pretend to refer to the objects. This may seem to be a "clean" proposal ontologically, but it gives no mechanism to account for either the semantic coherence of the text or the grammatical structure of its sentences.

If one simply pretends to do something it is not clear that there is any structural residue from the pretense. There is nothing in a pretense that would connect the events on the author's time-line with any events in the fiction. Even Searle (1975a) agrees that after fictional narrative is written, there are fictional characters and ordered fictional events created by the author which can be referred to in a discussion of the fiction. The author must have done something instead of merely pretended to do something. What she has done is created the fiction, that is, created a possible world which has space, time, events, existents, characters, etc. (Chatman, 1978).

Fictional Narrative Text. Either simultaneous with the creation of the fiction or subsequent to it, the author writes sentences which express some components of this fiction. If she uses sentences with finite verbs, to the extent that the verbs are deictic, the author linguistically puts an image of the creator of the sentences somewhere on the fictional time line. By generating those sentences the author "pretends" that she is somewhere in the fictional world. This claim is consistent with Searle (1975a). After quoting from Sherlock Holmes a passage in which Watson is beginning to tell of an adventure, Searle writes, "Sir Arthur is not simply pretending to make assertions, but he is pretending to be John Watson, M.D....making assertions" (p. 328, Searle's emphasis). Watson is on the fictional time line, and if Sir Arthur is pretending to be him, he projects an image of himself into Watson. In that way Sir Arthur can meaningfully contextualize the sentences ostensively uttered by Watson.

Any sentence telling or presenting the story, by the nature of the language, must be expressed from somewhere along the time-line of the story. This is done by the author contextualizing the sentence from

somewhere within the world of the story. A point somewhere on that fictional time-line serves as the temporal source of the sentence. Where on that line, is one of the constraints on the selection of tense and other text. The sentences of the text cannot be contextualized as a story without being on the time-line somewhere. Note that this is a requirement that the sentences are contextualized from within the text, not that they are expressed by a fictional narrator.

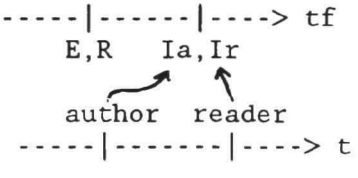
The Reader. The role and location of the reader of fiction has to be clarified. Although speech act theory has generally identified the participants of a speech act as speaker and hearer, the hearer has usually been conceived of as the addressee (cf. Clark & Carlson, 1982). Clark and Carlson have argued that there are participants in receiving language besides the addressee, and that an informative is one component of all speech acts. There is evidence that language is often constructed for participants other than direct addressees. Jimmy Swaggart addressed the confession of his sins directly to God before several million people on television. Conversations over mass media are designed to be heard by non-addressees. Clark and Carlson give the example

(11) Othello, to Desdemona, in front of Iago and Roderigo: Come Desdemona

Iago and Roderigo were informed of the request, as are millions of readers and viewers of Othello.

There are a few sentences in fictional narrative in which the author uses sentences which are directed at the reader, but most sentences are not. The reader is often a silent participant, seeing the sentences, but not being addressed by them (cf. Banfield, 1982).

The author neither has to write through a narrator nor to address the reader in constructing the sentences of the story. She simply has to present the story in the form of text from a position within the story world. The sentences are thus contextualized from a particular position in that world. The reader, in order to understand the set of relations expressed, must also be deictically connected to the text. He can best do that by assuming a location on the time-line and in the story world identical to or near that from which the sentences are contextualized. He does this by shifting his Deictic Center conceptually to this point. As the author writes by pretending that she is at a particular contextual location in the story, the reader reads by pretending that he is at the same place. Since the reader understands the story from the perspective of his image in the world of the story, it does not matter where he "really" is situationally. The same holds true for the author. What matters is where they are imaginally or conceptually. Diagrammatically, a fictional past-tense sentence may exhibit these structural relations:

(12)


The image of the author and the image of the reader may be at the same place and time fictionally, regardless of their temporal and spatial relationship in the real world.

Applications. This construal can incorporate much of the work done in the analysis of fictional text and suggests a direction for solving many of its conceptual problems. Hamburger (1973), Cohn (1978), Genette (1980), Banfield (1982), Wiebe (1989), and many others, have shown that many sentences in fictional text are often presented using personal or expressive references to characters and other expressions of feelings, knowledge and belief which can only come from the subjectivity of the characters themselves. Weibe (1989; Weibe & Rapaport, 1988) has developed an algorithm that identifies "subjective contexts." These are contexts which cue the reader that the sentences following are to be contextualized in the subjectivity of a character which is currently on the top of a stack. Thus the reader understands the sentences by locating an image of himself within the subjective space of the character.

There are many discussions of tense. Casparis (1975) finds many examples of the present tense in fiction. However, his model requires a narrator, and often no narrator is at the scene, so Casparis argues that tense does not mark time. Since the Deictic Shift Model give the author an ability to establish a source anywhere within the story she can be located at the time of the scene to express the sentences from that perspective.

Almeida (1987) has written a developed a model which understands the temporal continuity of fictional discourse by showing how the NOW of the text is often sequentially updated so that the sentences that successively appear in the text, even though they are not structurally tied can be seen to be integrated temporally. This is done by establishing the reader on the fictional time line of the narrative.

This paper is an informal presentation of the idea that all fiction is deictic. This idea seems very rich in its possibilities, although there are many problems to be solved, some of which may turn out to be traps.

References

Banfield, A. (1982). Unspeakable Sentences: Narration and Representation in the Language of Fiction. Boston: Routledge & Kegan Paul.

Barwise, J. and Perry, J. (1983). Situations and Attitudes. Cambridge, MA: MIT Press.

Booth, W. (1983). The Rhetoric of Fiction, 2 ed. Chicago: University of Chicago Press.

Bruder, G.A., Duchan, J.F., Rapaport, W.J., Segal, E.M., Shapiro, S.C., and Zubin, D.A. (1986) Deictic Centers in Narrative: An interdisciplinary Cognitive-Science Project, Tech. Report 86-20, Buffalo: SUNY Buffalo Dept. of Computer Science.

Buhler, K. (1934/1982). The deictic field of language and deictic words In R.J. Jarvella and W. Klein (Eds.) Speech, Place and Action: Studies in Deixis and Related Topics. Chichester, England: Wiley.

Casparis, C. P. (1975). Tense Without Time: The Present Tense in Narration. Zurich: Francke Verlag Bern.

Chatman, S. (1978). Story and Discourse. Ithaca: Cornell University Press.

Clark, H.H. and Carlson, T.B. (1982). Hearers and speech acts, Language, 58, 332-373.

Cohn, D. (1978). Transparent Minds, Princeton: Princeton University Press.

Fillmore, C.J. (1981). Pragmatics and the description of discourse. In P. Cole (Ed.) Radical Pragmatics, New York: Academic Press.

Galbraith, M. (1989). Subjectivity in the Novel: A Phenomenological and Linguistic Approach to the Narration of Childhood Self. Unpublished Ph.D. Dissertation, SUNY Buffalo.

Genette, G. (1980). Narrative Discourse: An Essay in Method, translated by J.E. Lewin, Ithaca: Cornell University Press.

Genette, G. (1988). Narrative Discourse Revisited, translated by J.E. Lewin, Ithaca: Cornell University Press.

Hamburger, K. (1973). The Logic of Literature, translated by M.J. Rose, Bloomington: Indiana University Press.

McHale, B. (1983). Unspeakable sentences, unnatural acts, linguistics and poetics revisited, Poetics Today, 4, 17-45.

Rapaport et al., (1989). Deictic Centers and the Cognitive Structure of Narrative Comprehension, Tech. Report 89-01, Buffalo: SUNY Buffalo Dept. of Computer Science.

Reichenbach, H. (1947) Elements of Symbolic Logic, New York: Dover.

Schopf, A. (Ed.) (1987, 1989) Essays on Tensing in English, (Two Volumes) Tubingen: Niemeyer.

Searle J. (1969). Speech Acts, Cambridge: Cambridge University Press.

Searle J. (1975a). The logical status of fictional discourse, New Literary History, 6, 319-332.

Searle J. (1975b). A taxonomy of illocutionary acts. In K. Gunderson (Ed.) Language, Mind, and Knowledge, Minnesota Studies in Philosophy of Science, Vol. VII. Minneapolis: University of Minnesota Press.

Wiebe, J.M. (1989). Recognizing Character's Thoughts and Perceptions: A Computational Investigation of Narrative Text, Unpublished Ph.D. Dissertation, SUNY Buffalo.

Wiebe, J. and Rapaport, W.J. (1988). A computational theory of perspective and reference in narrative, Proc. 26th Annual Meeting of the Association of Computational Linguistics, 131-138.

THEMATIC ROLES AND PRONOUN COMPREHENSION[1]

Rosemary J. Stevenson, Rosalind A. Crawley, Garry Wilson and David Kleinman
Department of Psychology
University of Durham
England

ABSTRACT

Two experiments tested the view that thematic role information triggers the rapid retrieval of general knowledge in pronoun comprehension. Pairs of thematic roles were contrasted as antecedents of a subsequent pronoun. The results showed that intepretation of the pronoun depended on the thematic role of the antecedent. Experiment one measured reading rates for the clause which contained the pronoun. Rates were faster when the antecedent was an Agent subject, a Patient object, a Goal, or an Experiencer. Rates were slower when the antecedent was an Agent object, a Patient subject, a Source, or a Stimulus. Experiment two required subjects to write completions to sentence fragments such as Jill like Sue and she..... and the number of references to each antecedent was recorded. The results confirmed the findings from Experiment one, although there was also an antecedent position effect (first vs. second mention) in some of the sentences. We suggest that these results are consistent with the view that thematic role information triggers the retrieval of canonical events in the real world, and may thus be responsible for the rapid retrieval of general knowledge in language comprehension.

INTRODUCTION

A striking feature of pronoun comprehension is its reliance on inferences from general knowledge (e.g. Stevenson and Vitkovitch, 1986). Some accounts propose that general knowledge may be accessed in the form of scripts (e.g. Sanford and Garrod, 1981), but scripts may not be sufficiently flexible for the fast retrieval of general knowledge (e.g. Kintsch, 1988). Others have used mental models (e.g. Garnham, 1989; Johnson-Laird, 1983). While the notion of mental models provides a useful theoretical framework, it does not explain how general knowledge is accessed when a mental model is constructed.

One possible mechanism for the fast retrieval of general knowledge when comprehending pronouns is the use of thematic role information. It has been argued that thematic roles may provide a mechanism for mediating between parsing, a discourse model and real world knowledge (e.g. Carlson and Tannenhaus, 1988; Rayner, Carlson and Frazier, 1984). However, the relationship between conceptual roles and grammatical roles is not straightforward (e.g. Finer and Roeper, 1989). Furthermore, there is no general consensus on the theoretical status of thematic roles (cf. Ladusaw and Dowty, 1988; Williams, 1989), but what is clear is that people do categorize the world in terms of agents, patients, experiencers and so on (e.g. Brown and Fish, 1983), and that these categories are also encoded in the language. Thus they may provide a convenient interface between language and general knowledge so that the rapid retrieval of general knowledge is possible, as suggested by some of the results discussed by Carlson and Tanenhaus (1988).

The hypothesis we wish to test is that there are canonical events in the world and that the lexical information about thematic roles in the verb triggers the retrieval of these canonical events. If the situation described by the sentence corresponds to the triggered canonical event, then comprehension is fast, otherwise comprehension is slowed. For example, action verbs trigger the retrieval of a canonical agent-patient sequence (e.g. Pinker, 1989). Thus, sentences containing such verbs should be comprehended more rapidly if they describe agent-patient sequences rather than patient-agent sequences.

An alternative conception of the way thematic roles might be used is the notion of a thematic hierarchy. Thematic hierarchies have been proposed by Jackendoff (1972) and Nishigauchi (1984). In Jackendoff's hierarchy, Agent is higher than Locative, Source and Goal, which in turn are higher than Theme (or Patient). Nishigauchi proposes in addition that Goal is higher than Location or Source. These hierarchies might reflect preferences in the conceptual system, and this would lead us to predict that the higher a role in the hierarchy, then the more rapidly it is retrieved during comprehension.

EXPERIMENT ONE

We tested these views by measuring reading rates for clauses containing pronouns where the thematic roles of the antecedents were varied. We used sentences consisting of two clauses. The first clause contained two noun phrases, one of which was the antecedent of a pronoun that was contained in the second clause. Our sentences were of four main types. Three contained thematic roles from Nishigauchi's hierarchy: Agent-Patient sentences, Goal-Source sentences, and Agent-Goal/Source sentences. These latter sentences were included to assess the possibility that pronoun assignment in the Goal-Source sentences is dependent on the transfer of goods, and they each contained a verb of motion in the first clause. Finally, we used Experiencer-Stimulus sentences. Examples of the four types of sentences are shown in Table one. The examples in Table one all contain linguistically ambiguous pronouns: the two individuals mentioned in the first clause are of the same gender. However, the content of the second clause biased the interpretation of the pronoun to one of the two potential antecedents. These materials were constructed and assessed by 5 judges, initially independently and then in group discussion.

On the basis of the view that thematic role information triggers the retrieval of canonical events, we predicted that pronouns would be interpreted more rapidly when their antecedents were Agents in first (subject) position and Patients in second (object) position. We had no strong predictions for the other sentence types. But a plausible view is that in Goal-Source sentences, Goals make better antecedents than Sources because the Goal is in possession of the Theme at the time of the event described by the second clause (see also Ladusaw and Dowty, 1988). In the Agent-Goal/Source sentences, the Agent is more likely to be carrying out a second event, although this prediction is confounded with the position of the antecedent. Experiencer-Stimulus sentences seem to lead people to attribute the cause of the experience to the Stimulus (e.g. Brown and Fish 1983), which suggests that people do categorize events in terms of Experiencer and Stimulus. In our Experiencer-Stimulus

sentences, the second clause was a consequence or elaboration of the experience described in the first clause (see Table one). Our general knowledge of real world situations might lead us to expect, therefore, that this additional information also concerns the Experiencer. So when the pronoun has an Experiencer antecedent it should be comprehended more rapidly than when it has a Stimulus antecedent.

Agent-Patient Sentences

Pronoun biased to Agent antecedent
Joseph hit Patrick and he made sure that it hurt.
Patrick was hit by Joseph and he made sure that it hurt.

Pronoun biased to Patient antecedent
Patrick was hit by Joseph and he began to cry very loudly.
Joseph hit Patrick and he began to cry very loudly.

Goal-Source Sentences

Pronoun biased to Goal antecedent
Sarah borrowed a record from Jenny and she listened to it that evening.
Jenny lent a record to Sarah and she listened to it that evening.

Pronoun biased to Source antecedent
Jenny lent a record to Sarah and she asked for it back again.
Jenny borrowed a record from Sarah and she asked for it back again.

Agent-Goal/Source Sentences

Pronoun biased to Agent antecedent
Phil drove towards Len and he braked at the last minute. (Agent-Goal)
Len drove away from Phil and he crashed into a brick wall. (Agent-Source)

Pronoun biased to Goal or Source antecedent
Phil drove towards Len and he jumped out of the way. (Agent-Goal)
Len drove away from Phil and he waved at the disappearing car. (Agent-Source)

Experiencer-Stimulus Sentences

Pronoun biased to Experiencer antecedent
Darren disliked Martin and he made it clear to everyone.
Martin annoyed Darren and he stormed out of the room.

Pronoun biased to Stimulus antecedent
Martin annoyed Darren and he regretted it later that evening.
Darren disliked Martin and he reciprocated the ill will entirely.

Table one: Examples of the materials used in Experiment one. The underlined noun phrase is the intended antecedent, and is mentioned either first or second in the sentence.

From Nishigauchi's hierarchy, we predicted that pronouns should be interpreted more easily when their antecedents were Agents rather than Patients, Goals rather than Sources and Agents rather than Goals or Sources. The thematic hierarchy makes no prediction about the Experiencer-Stimulus sentences.

There were four factors in the experiment: the four sentence types; the two thematic roles in each sentence; position of the antecedent in the first clause (either first or second mentioned); and type of noun phrase, either a name, as in Table one, or a definite description (e.g. the policeman). There were two sentences in each of these 32 experimental conditions.

A self-paced reading time task was used. Sentences appeared one clause at a time on a computer screen and subjects pressed the space bar of the computer keyboard when they had read and understood each clause. The time taken to read the second clause of each sentence was measured. Questions were asked after approximately one third of the sentences to ensure that they were read for comprehension. Sixty four filler sentences were also included which tested hypotheses about temporal expressions. The results are shown in Table two. Since there was no effect of type of noun phrase and nothing interacted with this factor, the data have been combined for this factor.

	Position of antecedent in the sentence		
Thematic role of antecedent	First Mentioned	Second Mentioned	Overall Means
Agent	5.07	4.71	4.89
Patient	4.77	5.09	4.93
Goal	5.65	5.68	5.66
Source	5.00	5.33	5.16
Agent-Goal	5.16	5.16	5.16
Agent-Source	5.24	4.77	5.01
Experiencer	5.38	5.12	5.20
Stimulus	4.87	4.70	4.78

TABLE TWO: Mean reading rates (in words per second) for the second clause in each antecedent condition for each sentence type.

The main findings were as follows: the clause containing the pronoun was read more quickly when the antecedent was Agent in first position and Patient in second position. This is consistent with the retrieval of a canonical Agent-Patient event, but inconsistent with the thematic hierarchy which predicted that the clause should be read more quickly when the antecedent was Agent rather than Patient, regardless of its sentence position. In the Goal-Source sentences, the clause was read more quickly when the antecedent

was Goal rather than Source, as both the hierarchy and the canonical event hypotheses would predict. In the Agent-Source sentences, the clause was read more quickly when the antecedent was the Agent rather than Source. This is only partially consistent with the two hypotheses, which both predict a similar preference in Agent-Goal sentences. In the Experiencer-Stimulus sentences, the clause was read more rapidly when the antecedent was Experiencer rather than Stimulus, consistent with the view that the verbs trigger the retrieval of canonical events. The thematic hierarchy makes no predictions for such sentences.

The details of the analyses are as follows: Agent-Patient sentences showed an interaction between thematic role and position of the antecedent ($F1=7.00$, $df=1,31$, $p<.05$; $F2=5.31$, $df=1,15$, $p<.05$). Reading rates were facilitated when the Agent was mentioned first and when the patient was mentioned second. Goal-Source sentences showed a main effect of thematic role ($F1=18.45$, $df=1,31$; $p<.01$; $F2=15.62$, $d=1,15$, $p<.01$). Goal antecedents were retrieved more quickly than Source antecedents. Agent-Goal/Source sentences showed a main efect of position of antecedent ($F1=6.89$, $df=1,31$, $p<.05$; $F2=3.28$, $df=1,15$, $p<.1$). Reading rates were facilitated when the antecedent was the Agent, and hence mentioned first. There was also a significant interaction between thematic role and antecedent position in the F1 analysis only ($F1=4.47$, $df=1,31$, $p<.05$; $F2=2.02$, $df=1,15$,). Facilitation for the Agent was only apparent when the Agent occurred with a Source. Experiencer-Stimulus sentences showed a main effect of thematic role ($F1=11.17$, $df=1,31$, $p<.01$; $F2=7.94$, $df=1,15$, $p<.05$). Pronouns were interpreted more quickly when the antecedent was Experiencer rather than Stimulus. There was also a main effect of position of the antecedent, in the F1 analysis only ($F1=4.33$, $df=1,31$; $p<.05$; $F2=2.76$, $df=1,15$). Reading rates were facilitated when the antecedent was mentioned first rather than second in the sentence.

These results were replicated in an additional experiment using modified materials, which included linguistically unambiguous pronouns: the two individuals mentioned in the first clause were different genders. The results confirmed the main findings above with two exceptions: (1) the facilitating effect of the Agent appeared in both Agent-Goal and Agent-Source sentences, and (2) the facilitating effect of the Experiencer antecedent only appeared when it was the first mentioned antecedent. These results held for linguistically unambiguous pronouns as well as ambiguous ones.

EXPERIMENT TWO

Experiment two extended the generality of the findings by using a sentence continuation task. Sentences from the replication experiment were used. Subjects were presented with with sentence fragments up to and including the pronoun (e.g. Jill loathed Susan and she....), and were instructed to complete the sentence. The number of times the completions referred to either of the two potential antecedents was recorded. The results are shown in Table three. The main findings of Experiment one were confirmed. But the sentence continuation task produced additional antecedent position effects in the Agent-Patient and Experiencer-Stimulus sentences.

Thematic role of antecedent	Position of antecedent in the sentence: First Mentioned	Second Mentioned	Overall Means
Agent	3.00	0.63	1.81
Patient	7.34	5.00	6.17
Goal	6.66	5.72	6.19
Source	2.25	1.34	1.80
Agent-Goal	5.63	2.22	3.92
Agent-Source	5.75	2.34	4.05
Experiencer	6.66	5.50	6.08
Stimulus	2.47	1.31	1.89

TABLE THREE: Mean number of completions (out of 8) containing references to each antecedent in Experiment two.

Details of the analyses are as follows: Agent-Patient sentences showed a main effect of thematic role ($F1=93.27$, $df=1,31$, $p<.01$; $F2=59.24$, $df=1,15$, $p<.01$), and a main effect of antecedent position ($F1=22.44$, $df=1,31$, $p<.01$; $F2=50.42$, $df=1,15$, $p<.01$). Continuations referred to the Patient rather than the Agent and to the first mentioned antecedent rather than the second. Goal-Source sentences showed a main effect of thematic role ($F1=129.76$, $df=1,31$, $p<.01$; $F2=115.56$, $df=1,15$, $p<.01$). Continuations referred to the Goal. Agent-Goal/Source Sentences showed a main effect of antecedent position ($F1=13.16$, $df=1,31$, $p<.01$; $F2=15.71$, $df=1,15$, $p<.01$): Continuations referred to the first mentioned antecedent (the Agent) rather than to the second. Experiencer-Stimulus sentences showed a main effect of thematic role ($F1=135.50$, $df=1,31$, $p<.01$; $F2=166.26$, $df=1,15$, $p<.01$), and a main effect of antecedent position ($F1=4.03$, $df=1,31$, $p<.06$; $F2=14.93$, $df=1,15$, $p<.01$). Continuations contained more references to the Experiencer than to the Stimulus and more references to the first mentioned antecedent than to the second.

Thus the sentence continuation task also reveals an effect of thematic roles. There was a preference to refer to (1) Patients rather than Agents, (2) Goals rather than Sources, (3) Agents rather than Goals or Sources and (4) Experiencer antecedents rather than Stimulus antecedents. In addition there was a preference to refer to the first mentioned antecedent in all sentence types except Goal-Source.

DISCUSSION

The clearest evidence for the proposal that verbs retrieve canonical real world events comes from the Agent-Patient sentences in Experiment one: clauses containing pronouns were read more rapidly when the antecedent was an Agent

subject (mentioned first) or a Patient object (mentioned second). At first glance, the results of the continuation task seem to contradict this hypothesis because the continuations referred to Patients rather than Agents in both antecedent positions. However, there was also a first mention effect, which suggests that the continuations may be sensitive to topicalization in the passive form. That is, over and above a preference for Agent subjects and Patient objects, there is an additional preference for references to first mentioned noun phrases in the passives, i.e. to Patient subjects. However, the results for Agent-Patient sentences from both experiments are inconsistent with the hierarchy hypothesis, which predicts an overall preference for the Agent rather than the Patient. There is an alternative interpretation of the Agent-Patient results of Experiment one. This is that the reading rates are due to a lack of parallelism in the sentences with passive first clauses (see e.g.Frazier, Taft, Roeper and Clifton, 1984). Unfortunately it is difficult to construct Agent-Patient sentences where the canonical event hypothesis is not confounded with a parallelism hypothesis. However, inspection of the continuations from Experiment two reveals that people frequently produce non-parallel structures when the Patient is mentioned first.

The results from the Goal-Source sentences endorse the importance of thematic roles without distinguishing between canonical events and a thematic hierarchy. Reading rates are fast when the sentences describe situations where the person in possession of the goods is also the actor of the second event. In the continuations, the person who possesses the goods becomes the actor of the second event. The results from the Agent-Goal/Source sentences are also consistent with both hypotheses: the Agent being preferred to both Goal and Source. Finally, the results from the Experiencer-Stimulus sentences again endorse the importance of thematic roles and are consistent with the canonical event hypothesis. The preference for Experiencer antecedents in both experiments can only be explained by reference to real world events.

Thus the canonical event hypothesis explains more of the data than does the hierarchy hypothesis. We suggest, therefore, that one way in which general knowledge is rapidly retrieved for the comprehension of pronouns is by the lexical information in the verb triggering the retrieval of canonical events in the real world. The more closely the situation described by the sentence maps onto the canonical event, the more speedily will the pronoun be interpreted. These suggestions are not conclusive, but they do indicate how thematic roles might be used for the rapid retrieval of general knowledge. This use of thematic roles is likely to be modified by other information in the sentence, as shown by the topicalizing effect of the passive form. In addition it is likely that the verb information is used in conjunction with the connnective. In these experiments we used and. If we had used because the precise pattern of results would probably be different, particularly for the Experiencer-Stimulus sentences (e.g. Garvey, Caramazza and Yates, 1976). Nevertheless, if the canonical event hypothesis is correct, then it should contribute to a general theory of the way general knowledge is retrieved in comprehension and not just apply to pronouns.

REFERENCES

Brown, R. & Fish, D. (1983). The psychological causality implicit in language. Cognition, 14, 237-273.

Carlson, G.N. & Tanenhaus, M.K. (1988). Thematic roles and language commprehension. In W. Wilkins (Ed.), Syntax and Semantics, Volume 21. San Diego: Academic Press.

Finer, D. & Roeper, T. (1989). From cognition to thematic roles: the projection principle as an acquisition mechanism. In R.J. Mathews & W. Demopoulos (Eds.), Learnability and Linguistic Theory. Dordrecht: Kluwer Academic Publishers.

Frazier, L., Taft, L. Roeper, T. & Clifton, C. (1984). Parallel structure: a source of facilitation in sentence comprehension. Memory and Cognition, 12, 421-430.

Garnham, A. (1989). Mental Models as Representations of Discourse and Text. Chichester, England: Ellis Horwood/John Wiley and Sons.

Garvey,C., Caramazza, A. & Yates, J. (1976). Factors influencing assignment of pronoun antecedents. Cognition, 3(3), 227-243.

Jackendoff. R.S. (1972). Semantics in Generative Grammar. Cambridge, Mass.: MIT Press.

Johnson-Laird, P.N. (1983). Mental Models. Cambridge, Mass.: Harvard University Press.

Kintsch, W. (1988). The role of knowledge in discourse comprehension: a construction-integration model. Psychological Review, 95, 163-182.

Ladusaw, W.A. & Dowty, D.R. (1988). Toward a nongrammatical account of thematic roles. In W. Wilkins (Ed.), Syntax and Semantics, Volume 21. San Diego: Academic Press.

Nishigauchi, T. (1984). Control and the thematic domain. Language, 60, 215-250.

Pinker, S. (1989). Learnability and Cognition: The Acquisition of Argument Structure. Cambridge, Mass., London, England: Bradford/MIT Press.

Rayner, K., Carlson, M., & Frazier, L. (1984). The interaction of syntax and semantics in sentence processing: eye movements in the analysis of semantically biased sentences. Journal of Verbal Learning and Verbal Behaviour, 22, 358-374.

Sanford, A.J. & Garrod, S.C. (1981). Understanding Written Language. Chichester, England: John Wiley & Sons.

Stevenson, R.J. & Vitkovitch, M. (1986). The comprehension of anaphoric relations. Language and Speech, 29, 335-360.

Williams, E. (1989). The anaphoric nature of O-roles. Linguistic Inquiry, 20, 425-456.

[1] This research was supported by Grant No. RC0023 2441 from The Economic and Social Research Council, Great Britain to the first two authors, and by the Human Communication Research Centre of the ESRC, Great Britain at the Universities of Edinburgh, Glasgow and Durham.

Parallel Processes During Question Answering?

Scott P. Robertson and Kimberly C. Weber

Psychology Department
Rutgers University
Busch Campus
New Brunswick, NJ 08903

Abstract

Question answering involves several processes: representation of the question concept, identification of the question type, memory search and/or inference generation, and output. Researchers tend to view these processes as stages and have developed primarily serial models of question-answering. Word-by-word reading times of questions, however, suggest that some processing is done in parallel. Questions were read more slowly but answered quicker when the question type was apparent from the first question word (the usual English construction of a question) when compared to cases when the question word came last. Serial models can not explain such data easily. It is argued that the processes associated with a particular question type are active during processing of the question concept and that they can direct memory search during question parsing. Some parallel models of question answering consistent with the data are discussed.

Introduction

Question answering is a process that requires parsing, directed memory search, sometimes inference generation, and finally the formulation of an answer. The processes involved in many of these steps have been studied extensively using methods from both artificial intelligence (Lehnert, 1978) and cognitive psychology (Singer, 1984, 1986; Graesser, Robertson, & Anderson, 1981). Not much is known, however, about the time course of these steps and which components may be realized in parallel. In this study the hypothesis that question parsing and memory search are integrated, parallel processes is assessed.

Lehnert (1978) suggested that the process of question answering involves representation of the *question concept* and categorization of the *question type*. Once these steps are completed, information about the question type can be used to initiate appropriate memory search or inference procedures while the question concept serves as the indexing mechanism to the memory representation. For example, questions 1 and 2 below both contain the question concept *Pete drove to the store*, but they are two different question types.

(1) Did Pete drive to the store?
(2) Why did Pete drive to the store?

Question 1, a *verification* question, may be answered simply by finding the question concept in memory. The simplest procedure associated with verification specifies that if the question concept is present in memory then the answer to a verification question is "true," otherwise it is "false" (Singer, 1984). More complex models of verification, the use of plausibility criteria for example, have also been proposed (Reder, 1982). In any case, matching processes require explicit representation of the question concept in order to begin.

Question 2, a *goal* or *causal antecedent* question, requires complicated processing both to determine the question type and to retrieve an answer. The question concept must first be generated, but the next step for such questions depends on the structure of information associated with the concept. For example, if the subject of the question is an animate actor then the question is a *goal* question and memory retrieval will focus on the actor's goals. In such a case, if an explicit goal is associated with the concept, e.g. "Pete needed bread," then that goal would be a candidate for an answer; otherwise a more generic goal or preceeding state, e.g. "Pete was hungry" or "Pete was out of food," might be generated by inference as possible answers. If the subject of the question is

not animate, e.g. "Why did the tree fall," then the question is a *causal anecedent* question and memory retrieval and inference processes will not involve goals but will focus on preceding, causally relevant states.

In several studies Singer (1984, 1986) has examined both verification questions and "wh- questions" (who, what, when, and where) and used a stage model to explain question-answering times. The model assumes, like Lehnert, that conceptual representation of the question occurs first, followed by question categorization, followed by memory search, finally terminating in an output stage. In Singer's studies the time to read and answer questions in which the question word was the first word of the question sentence was measured. Singer's serial model predicted the overall time taken to answer the questions, but did not reveal processes that might be occurring during reading.

To compare serial versus parallel possibilities for question answering, consider the wording of questions 3 and 4 below.

(3) Why did Pete drive to the store?
(4) Pete drove to the store, why?

Most models of question answering assume that a conceptual representation of the question is built from the sentence before other processing begins. For both cases 3 and 4 we might designate such a representation with embedded propositions as follows:

(REASON, ? (DRIVE, ACTOR:Pete, FROM:unspec, TO:Store))

If a concept is found in memory that matches the above pattern, then the portion that replaces the "?" will serve as an answer to the question. Note that it is necessary to generate the complete pattern in order to initiate matching processes. On this view the only reason to expect a difference in the time course of reading and answering between questions 3 and 4 would be if different processes were involved in generating the question concept. If question answering doesn't begin until the concept is built, then it should proceed in the same way in both cases.

An alternative view is that some retrieval processes can begin for question 3 before they do for question 4. On this view the question word, *why* in the example, triggers activation of rules (or processes) specific to the question type that can apply while the rest of the question is being parsed. This parallel process model suggests that question 3 could be answered more quickly than question 4, although extra processing occurring during reading of the main part of question 3 might slow the reading rate relative to the same parts of question 4.

Reading Behavior	Question position	
	Question-first	Question-last
Before	+ #### ##### ## ### #####	#### ##### ## ### ##### +
After	Why? Pete drove to the store.	Pete drove to the store. Why?

Table 1.

Appearance of the screen before and after the subject reads a question in the two question-position conditions.

In this study subjects read concepts from brief paragraphs and answered questions about them. In one condition the question type was known as the concept was being read, as it would be in question 3 above. In a second condition the question type was not known until after the concept had been read, as in question 4 above. Reading times for the sentences expressing the question concepts, the words indicating the question types, and the answer times were collected. An advantage in answering time when the question type was known in advance would be taken as evidence for parallel processing during reading of the question concept.

Method

Subjects.
Twenty-two undergraduate students participated for course credit. Each subject served in all conditions of the experiment.

Materials.
Twenty five- to eight-line stories were written. All stories were about a character who performed several actions. An action was chosen to be queried from each story. Each action statement chosen for a query had the same syntactic form: NOUN1 + VERB + PREPosition + DETerminer + NOUN2.

Design and Procedure.
Subjects read the twenty stories in sequence and answered a question after each story. Each story was presented as a whole on a computer screen and subjects spent as long as they wished reading the story. Subjects indicated that they had read a story by pressing a response key. When the key was pressed the story disappeared from the screen and was replaced by a *question frame*. The question frame consisted of two lines, one containing a plus-sign where the question word would appear and the second containing number-signs where words expressing the question concept would appear as a sentence. In some cases the plus sign was above the sentence line indicating that the question word would appear first. In other cases it was below the sentence line indicating that the question word would appear last. Table 1 shows examples of the question frame as it appeared in the question-first and question-last conditions for a *why* query of the sentence "Pete drove to the store."

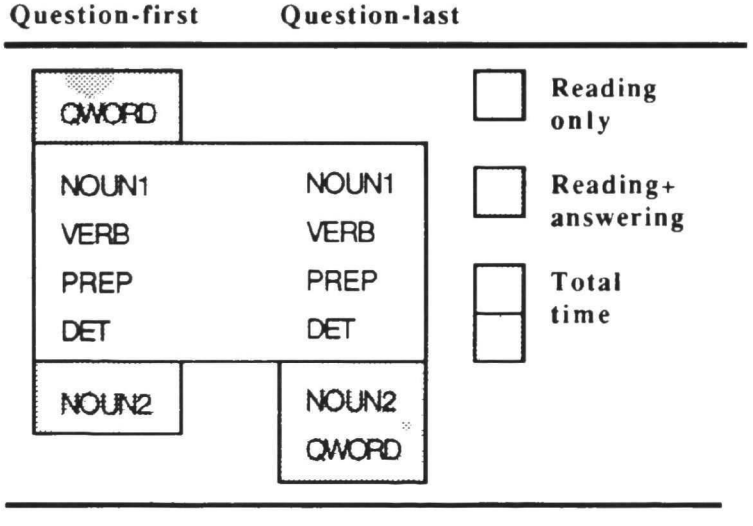


Figure 1.
A comparison of word positions and dependent measures in the two question-position conditions.

Subjects pressed the same response key repeatedly to reveal the question word (which replaced the plus sign) and each part of the question concept (words replacing the number signs). The reading time for each of these words was recorded. Subjects received special instructions for the last word (in some cases the last noun of the concept and in others a question word). They were instructed to think of the answer and press the response key as soon as the answer "came to mind." (A similar procedure was used succesfully by Reiser, Black, & Abelson, 1985, in studies of autobiographical memory.) Thus the reading time for the last word includes answer retrieval time. Subjects then wrote the answer on a sheet of paper and moved on to the next story at their own pace. All subjects received 20 practice trials to get used to the procedure.

Three types of questions were asked, *verification*, *goal*, and *time*. The question type was indicated by the presentation of the words *True?*, *Why?*, or *When?* in the question frame. Each subject answered all three types of questions (6 questions of each type) randomly intermixed. All of the 6 critical verification questions were affirmative, but two extra stories (not analysed) and several practice stories were followed by negative verification questions so that affirmation of this question type was not predictable.

There were two conditions related to the presentation position of the question words. In half of the trials the question word appeared before the concept sentence and in the other half it appeared after the concept sentence. Question type and question position were completely crossed and all subjects received stories in all combinations of these two factors. Stories were rotated through the question-type and question-position conditions across subjects. Presentation order of the stories, and hence the conditions, was random for each subject.

Results

Reading times were collected for the question words and each word expressing the question concept. Here the overall reading and answering time, the time to read the first four words of each sentence (combined and separately), and the time to read the question word plus the last word of each sentence are reported. These dependent measures are illustrated in Figure 1. As Figure 1 shows, the question-word *(QWORD)* reading time includes answering time in the question-last condition but not in the question-first condition and the *NOUN2*

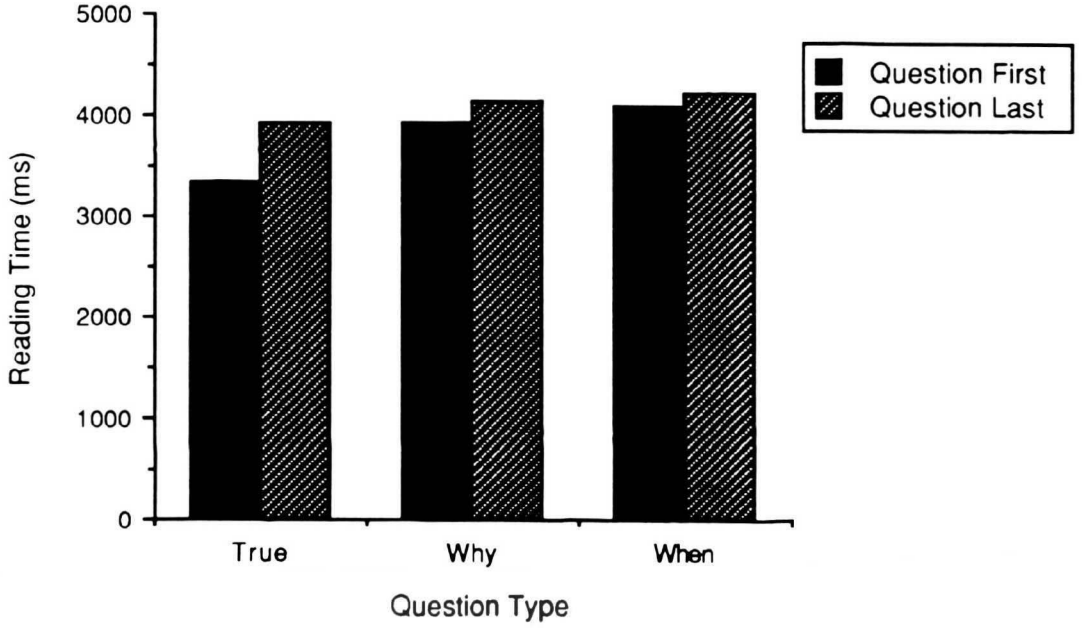


Figure 2. Total reading and answering time.

reading time includes answering time in the question-first condition but not in the question-last condition. By reporting *QWORD+NOUN2* in all conditions we are reporting the answering time combined with the (presumably constant) time to read each word separately.

Each subject provided three times in each of the six conditions. All reading times above 8000ms were eliminated. These times usually occurred when subjects forgot to press the reponse key before writing an answer. The mean times for as many good observations as a subject provided in each condition were used. In only one case was data missing because all three times from a subject in a particular condition were above the threshold. In this case the overall mean of that condition was used as a replacement value.

Figure 2 shows the overall mean reading and answering time for all words in the two question positions and across the three question types. The total time was shorter when the question word came first than when it came last, 3780ms versus 4090ms repectively, $F(1,21)=9.96$, $p<.01$, MSe=317,940. Also, there was a difference among the total times across question types, with the verification question being fastest, 3631ms, the why-question next, 4029ms, and the when question longest, 4145ms, $F(2,42)=3.42$, $p<.05$, MSe=937,618. There was no interaction between these factors.

Figure 3 shows the mean time to read *NOUN1+VERB+PREP+DET* combined in the two question positions and across the three question types. This time does not include answering time. The mean time to read this material was longer when the question was known, 1456ms versus 1355ms for the question-first and question-last conditions respectively, $F(1,21)=10.95$, $p<.01$, MSe=30,478. A question-type effect was not present for these words nor was an interaction.

Figure 4 shows the mean time per word for *NOUN1*, *VERB*, *PREP*, and *DET* separately in the two question-order conditions combined over question types. A word by question-position by question-type ANOVA showed no main effect of question type and no interactions of question-type with other factors. The apparent interaction between question position and word was significant $F(3,63)=3.84$, $p<.01$, MSe=2014. Individual comparisons revealed the suprising result that the question position effect was present for *NOUN1*, $F(1,21)=7.28$,

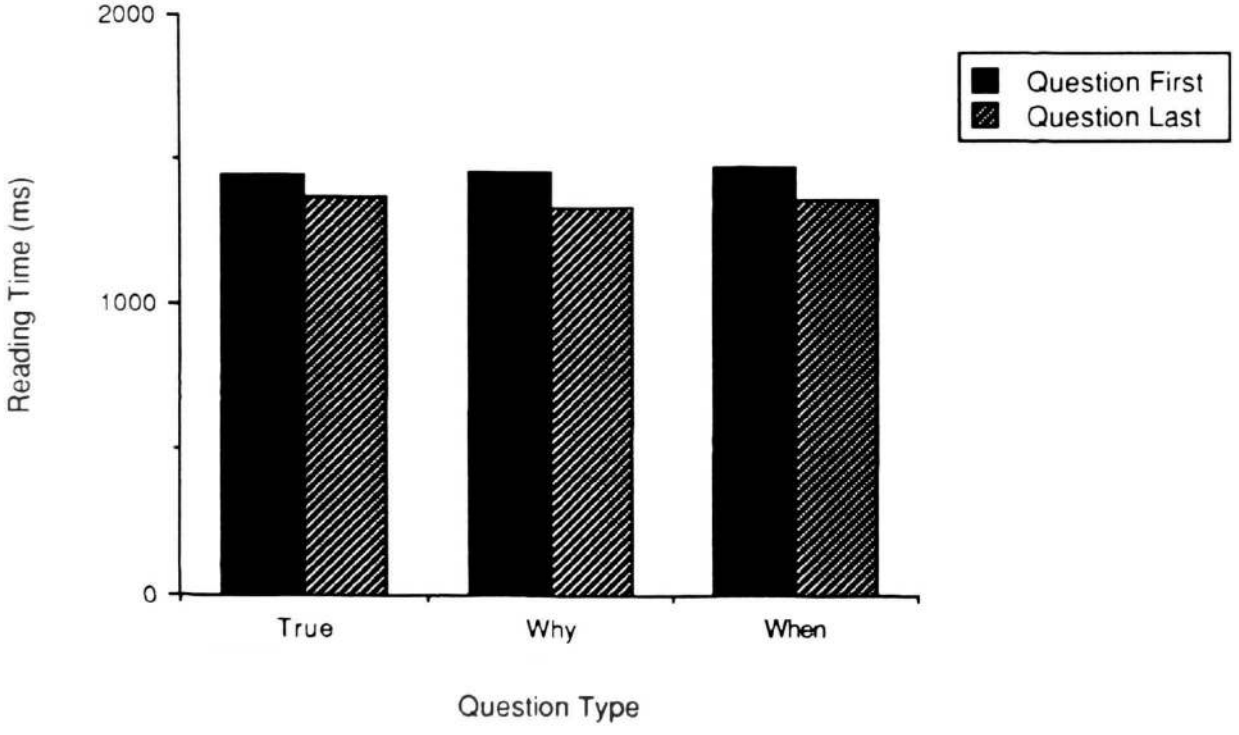


Figure 3. Reading time for NOUN1+VERB+PREP+DET.

$p<.01$, MSe=4489; *PREP*, $F(1,21)=5.96$, $p<.05$, MSe=3323; and *DET*, $F(1,21)=16.32$, $p<.01$, MSe=3651; but not for *VERB*. The means for each significant question-position effect were 367ms versus 336ms for *NOUN1*, 363ms versus 339ms for *PREP*, and 395ms versus 353ms for *DET*, in the question-first and question-last conditions respectively. There were no effects of question type and no interactions between question position and question type for individual words.

Figure 5 shows the mean times to read the question word plus the last noun *(QWORD+NOUN2)* in the two question positions and across the three question types. This time includes answering time. In this comparison the mean time was faster when the question was known, 2452ms versus 2769ms for the question-first and question-last conditions respectively, $F(1,21)=9.36$, $p<.01$, MSe=449,676. There was also a main effect of question type, with the verification question being fastest, 2163ms, the why question next, 2829ms, and the when question longest, 2840ms, $F(2,42)=4.48$, $p<.05$, MSe=1,316,768. There was no interaction between the two factors.

Discussion

The main hypothesis that the time to read and answer a question would be faster when the question type was known was confirmed. A reasonable explanation of this effect is that processes related to answer generation can be executed during parsing of the information in the question concept.

Separate analyses of times that included question-answering time versus those that did not showed that knowledge of the question actually increased reading times of words in the body of the question concept. The advantage in reading time when the question type was known came on (or after) the last word and more than made up for the earlier slower reading times. This strengthens the view of parallel processes since it suggests that resources involved in question processing and concept representation compete during reading to slow overall time but that by the end of the question the answer is much closer if the question was known.

The effects of question type have been observed before and were not suprising (Singer, 1884, 1986). While

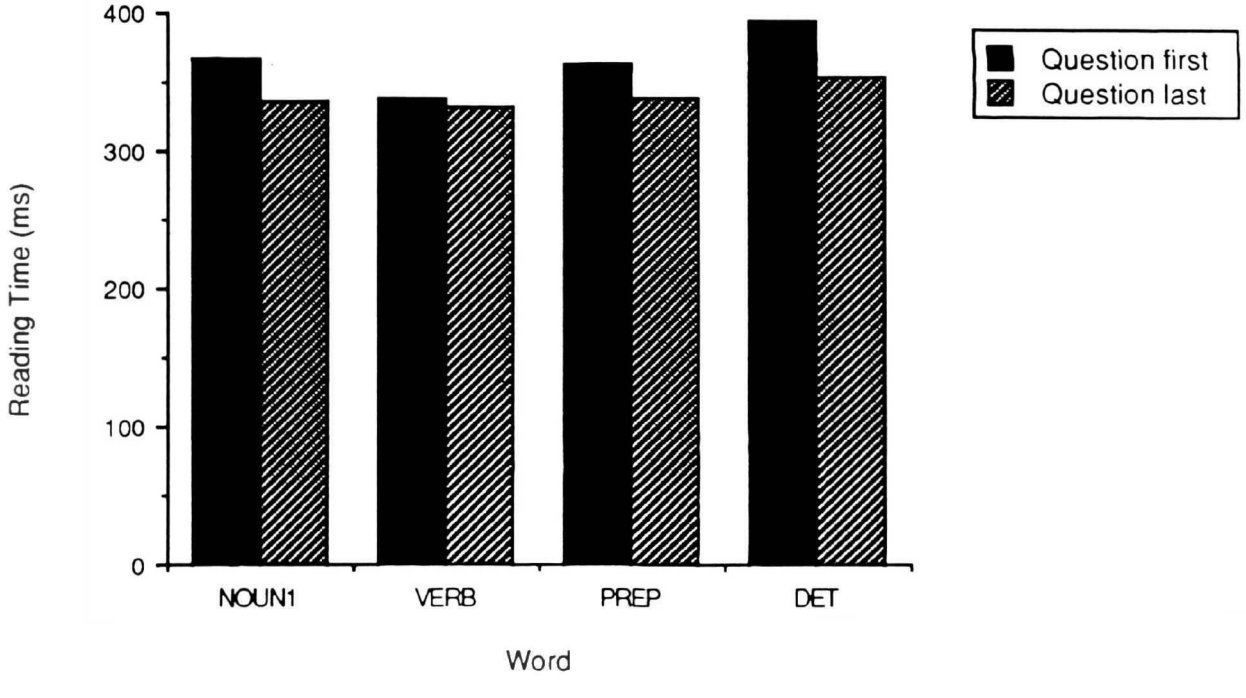


Figure 4. Reading time by word over all question types.

the mechanism behind this effect has never been specified, it is clearly related to the different operations that are required for answering different questions. For example, there are more potential answers to *goal* or *causal antecedent* questions than there are to *verification* questions, and so either memory retrieval time or output preparation time should be longer for the former types of questions. The fact that question-type effects appear only in the answering times, and not in the time for reading the main part of the sentence, may suggest that memory retrieval directed by processes specific to the question type does not occur in parallel with sentence comprehension. This would mean that processes common to all questions, preparation of question-answering processes for example, are what are occuring during comprehension. This phenomenon deserves further study.

These results rule out extreme serial models of question answering, e.g. Singer's (1981, 1984, 1986) VAIL models, retrieval mechanisms described by Graesser et al. (1981), or a strict interpretation of Lehnert's (1978) model. Such models hold that separate stages are involved in building the question concept, accessing the question-answering procedure, activating the question concept, and finally applying the procedure to the activated knowldge. The results may be explained, however, by different versions of parallel models.

In one parallel scenario the question word triggers proceeses that access the appropriate question-answering procedures. These procedures are assembled during reading, held in working memory during parsing of the question concept, and are then ready for application when the question concept is finally built. This model would explain why the question-position effect appears at the beginning of the sentence and why it disappears on the second word (once ready the question-answering procedures are not applied until later) but not why the effect *returns* toward the end of the sentence. Once question-answering procedures are present in working memory their effect on reading time should stay constant.

In a second parallel scenario question-answering procedures are accessed at the start of the question and begin to apply whenever there is enough information to do so. This model is consistent with the reappearance of the question-position effect later in the sentence (due to different retrieval processes). It is puzzling, however, under this model why the verb does not trigger parallel search processes since it clearly identifies the unique concept being queried. We are now repeating the experiment using different syntactic forms of the question

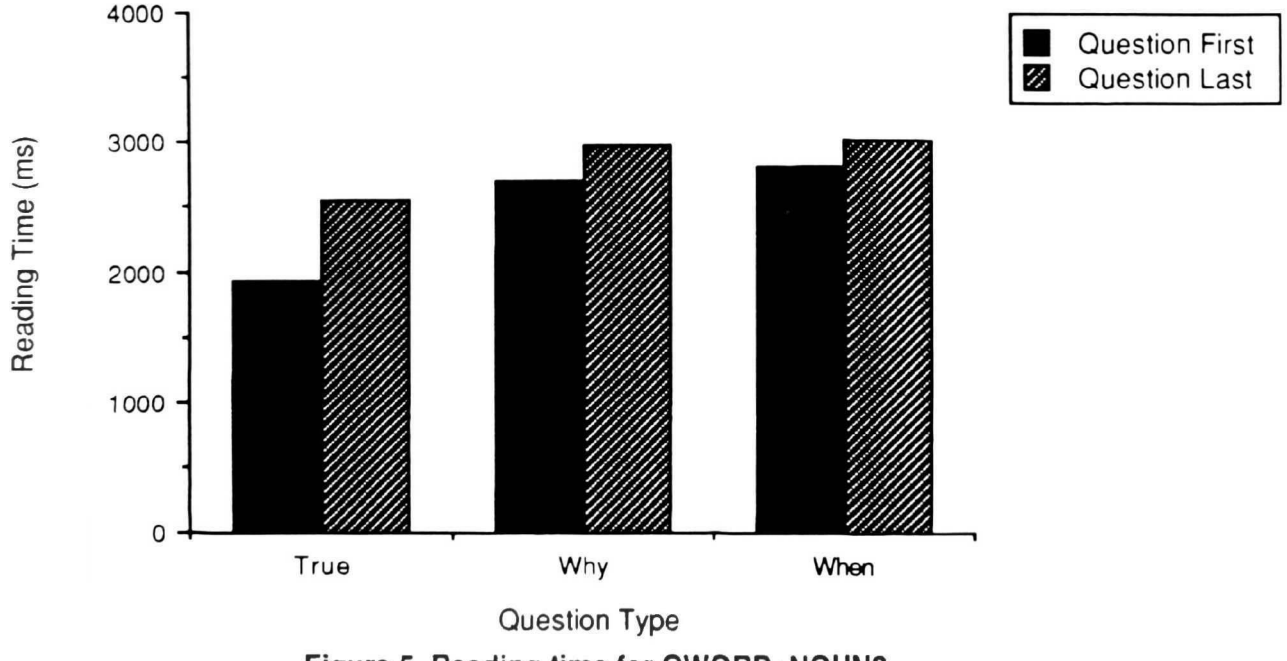


Figure 5. Reading time for QWORD+NOUN2.

concept in order to explore the intricacies of question processing during parsing.

Finally some comments should be made about symbolic versus connectionist models of this process. While the data indicate parallelism in question processing they should not be construed as evidence for or against a connectionist architecture. Wording such as "assembly of procedures for answering a why-question," a description from the point of view of a symbol-manipulation device, might easily be construed as "state of the network such that nodes related to goals and prior causal states are more likely to become active," a description from the point of view of a PDP device. It is not clear that one architecture is superior to another in explaining this data, although efforts to model the process should proceed along both lines.

To conclude it appears that question-answering processes and representation of the question concept can occur in parallel. This view conflicts with most current models of question answering. Details of a new model of these processes are now being developed. Also, further experiments to distinguish among various parallel algorithms are being conducted. Since language usually co-occurs with other cognitive processes, the ability to parallel process during language comprehension is important and the tradeoffs involved should receive careful attention from cognitive scientists.

References

Graesser, A.C., Robertson, S.P., & Anderson, P.A. (1981). Incorporating inferences in narrative representations: A study of how and why. *Cognitive Psychology, 13,* 1-26.

Lehnert, W.G. (1978). *The process of question answering.* Hillsdale, NJ: Erlbaum.

Reder, L.M. (1982). Plausibility judgements versus fact retrieval: Alternative strategies for sentence verification. *Psychological Review, 89,* 250-280.

Reiser, B.J, Black, J.B., & Abelson, R.P. (1985). Knowledge structures in the organization and retrieval of autobiographical memories. *Cognitive Psychology, 17,* 89-137.

Singer, M. (1981). Verifying the assertions and implications of language. *Journal of Verbal Learning and Verbal Behavior, 20,* 46-80.

Singer, M. (1984). Toward a model of question answering: Yes-no questions. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 10,* 285-297.

Singer, M. (1986). Answering wh- questions about sentences and text. *Journal of Memory and Language, 25,* 238-254.

Spreading Activation in PDP networks

James Hendler*
Computer Science Dept.
University of Maryland
College Park, Md. 20742
hendler@cs.umd.edu

Abstract

One argument in favor of current PDP models has been that the availability of "hidden units" allows the system to create an internal representation of the input domain, and to use this representation in producing output weights. The "microfeatures" learned by sets of hidden units, it has been argued, provide an alternative to symbols for certain reasoning tasks. In this paper we try to further this argument by demonstrating several results that indicate that such representations are formed. We show that by using a spreading activation model over the weights learned by networks trained via backpropagation, we can model certain cognitive effects. In particular, we show some results in the areas of modeling phoneme confusions and handling word-sense disambiguation, and some preliminary results demonstrating that priming effects can be modeled by this activation spreading approach.

1 Introduction

A primary difference between the neural networks of 20 years ago and the current generation of connectionist models is the addition of mechanisms which permit the system to create an internal representation. These "subsymbolic," semantically unnameable, features which are induced by connectionist learning algorithms have been discussed as being of import both in structured and distributed connectionist networks (cf. Feldman and Ballard, 1982; Rumelhart and McClelland, 1986). The fact that network learning algorithms can create these *microfeatures* is not, however, enough in itself to account for how cognition works. Most of what we call intelligent thought derives from being able to reason about the relations between objects, to hypothesize about events and objects, etc. If PDP models are to be used for cognitive modeling we must complete the story by explaining how networks can reason in the way that humans (or other intelligent beings) do.

To be able to claim that the internal representations learned by connectionist networks can provide a substrate on which such symbolic reasoning can be performed, we must be

*Also affiliated with the UM Institute for Advanced Computer Studies and the Systems Research Center. Partial support for this work was provided by the Office of Naval Research contract N00014-88-K-0560 and NSF grant IRI-8907890. This work was performed, in part, at the International Computer Science Institute in Berkeley, Ca.

able to demonstrate that this representation, *when removed from an input mapping,* can allow output concepts to be related together. Thus, for example, to account for the sorts of tasks which have motivated traditional AI models, such as natural language processing and planning systems, a system must be able to reason about "grandma" without having grandma (or someone who looks like her) physically available to the perceiver[1]. In a multi-layer PDP network, this minimally requires the ability to abstract relationships between the output units without directly activating input units.

In humans, this relatedness of concepts is usually tested via experimental paradigms that are based on a model in which activation spreads through an associative network relating concepts which are either perceptually or semantically linked. An examination of the spread of activation through semantically related concepts has been the focus of much work in categorization and lexical access[2]. Could this model, in which activation spreads between related concepts (represented by output concepts), be realized using the sorts of microfeature-based representations (encoded in the "hidden" units) learned by PDP models?

In this paper, we will demonstrate some evidence that this sort of model can, in fact, be realized in PDP models by using a direct analog of spreading activation. Essentially, activation at the output nodes is spread through the network of weights between hidden and output units in a three-layer network trained by the classical error back-propagation method. Using this technique, described in more detail in the next section, we are able to demonstrate that meaningful relationships between output units can be found. Following this we describe three results of this work: some results in the areas of modeling phoneme confusions (section 3) and handling word-sense disambiguation (section 4), and some preliminary results in demonstrating that priming effects can be demonstrated during this activation spreading (section 5).

It should be noted, however, that the evidence presented in this paper should not be taken as a direct model of human cognitive processing and activation spread. The differences between multiple-layer, back propagation trained, PDP networks and the either the hardware of the human brain or the human cognitive apparatus are many. All we wish to do in this paper is to demonstrate that distributed representations, such as those learned by these networks, could account for activation spreading effects. Thus, this work (and numerous extensions thereto) is *necessary* to demonstrating that a subsymbolic substrate could implement human-like cognition; however it is far from sufficient.

2 Activation Spread

Our system starts by assuming an activation pattern is started at one or more of the **output** nodes of the PDP network. This activation then spreads to the hidden unit and back to the output units. In this way, those units which share the most "microfeatures" in common

[1]Similarly, one can test this effect simply by closing one's eyes and thinking about "grandma" or any other symbolic entity.

[2]A complete list of citations is beyond the length limit on this paper. A long discussion of spreading activation models in AI and in psychology can be found in (Hendler, 1987; Chapter 8).

will gain the most activation[3].

The spread of activation between the output and hidden units in a PDP network is easily modeled. Consider a network, already trained, in which we have two output nodes, j and k. If j is activated with some energy, that energy will pass to each hidden node in proportion to the weight between j and that node. Each of these nodes, in turn, pass activation to the output nodes in proportion to the weights to those output nodes. Thus, k will gain activation from j given by:

$$v_{jk} = \sum_{i=1}^{I} w_{ji}.w_{ik},$$

where v_{jk} can be considered as a weight between output units k and i. In these networks, we treat the weights as symmetric, so that $v_{jk} = v_{kj}$. Where this symmetry holds, $V = [v_{jk}], (j,k = 1,...,N)$ (where N is the number of output units) is recognized to be, by definition, the mathematical covariance matrix of the weights between the hidden layer and the outputs. Thus, a traditional view of activation spreading, applied to these networks, is modeled by the well-behaved mathematical relationship of covariance[4]

While the covariance numbers are directly related to the patterns learned by the PDP network, they are unbounded, making them difficult to work with in modeling. As it is preferable to handle bounded numbers, in the experiments described in this paper, we replace the covariance matrix V by the mathematically "equivalent," although bounded matrix of correlation coefficients $C = [c_{kl}]$ computed as:

$$c_{kl} = \frac{\sigma_{kl}}{\sqrt{\sigma_{kk}.\sigma_{ll}}},$$

$$\sigma_{kl} = \frac{1}{I}\sum_{i=1}^{I}(w_{ik} - \overline{w_k})(w_{il} - \overline{w_l}),$$

$$\overline{w_k} = \frac{1}{I}\sum_{i=1}^{I} w_{ik},$$

and, consequently:

$$\begin{cases} c_{kk} = 1 \\ -1 \leq c_{kl} \leq +1. \end{cases}$$

The values c_{kl} are bounded and reflect the correlation, based on the parameters of the network, between output units k and l, $+1$ standing for the maximum positive correlation (i.e. virtually identical classes) and -1 for the maximum negative correlation (i.e. completely different).

In the remainder of this paper, we will examine what happens when this model of activation is applied to several specific networks. We will describe the training of the PDP models, and describe how the correlation-coefficient modeled, activation-spreading process can be used to show interesting aspects of the representations learned by PDP networks.

[3]This effect was first demonstrated in a hybrid system merging a local connectionist model and a symbolic marker-passer. Details of that work can be found in (Hendler, 1989).

[4]Others have modeled the spread of activation through a semantic memory using more complex functions. The best known of this work is the ACT* model discussed in Anderson (1983), which also presents a review of other systems using similar techniques.

3 Phoneme Confusion

Many phenomena needing to be explained by the cognitive scientist have their roots in the perceptual similarities between various objects as viewed by the human cognitive apparatus. Some examples include:

1. *Categorization*, where humans can classify objects as better and worse examples of some category (for example, "sparrow" is consistently rated as a better "bird" than "turkey").

2. Priming where the priming occurs for perceptually similar objects. An example of this is "rhyming priming," reported in numerous lexical access experiments, in which words which rhyme demonstrate priming effects.

3. Functional identification in which objects which have perceptual features in common are used to perform functions associated with each other (for example, substituting a rock for a hammer due to a similarity in mass).

Traditional symbolic modeling, in the AI and cognitive psychology communities, has been unable to offer an explanation of these effects.

In the human perceptual system, one of the "earliest" places in which similarity effects can be seen is in the perception of phonemes in continuous speech. Experimentation (Aubert, 1988) has shown that word confusions may arise from phoneme confusions occuring during speech perception. Thus, for example, the phoneme for the "short a" (Ahh) sound will be more likely to be confused with the perceptually related "short e" (ehh) sound than with the less related "hard g" (Guh) sound. Based on experimentation, Aubert produced a matrix of confusion likelihoods (Bounded from 0 to 1) between each of 50 phonemes he tested (thus producing a 50x50 correlation matrix between phonemes, based on human data). In his matrix, 194 cells had values different than 0 (no confusion) or 1 (the identity cells along the main diagonal of the matrix).

To see whether the activation spreading model described in section 2 would produce a similar matrix of phoneme confusions, an experiment was run in which a PDP network (with input, hidden, and output units) was trained to do phoneme identification. Using a technique developed by Bourlard, Morgan, and Wellekens (1989), a data base consisting of 100 sentences were used for training the network to recognize phonemes. Vector-quantized (132 prototypes) mel cepstra were used as acoustic vectors. To simplify the representation of the input data, each vector was replaced by its index coded by a simple binary vector with only one bit "on". Multiple frames were used as input to provide context (9 frames) to the network. Thus, the input field contained $9 \times 132 = 1188$ units, and the total of possible inputs was equal to 132^9. The size of the output layer was kept fixed at 50 units, corresponding to the 50 phonemes to be recognized. There were 26,767 training patterns representing only a small fraction of the possible inputs. A network with five hidden units was trained on this set, thus the network had 1188 inputs, 5 hidden, and 50 output units.

We compared the matrix of correlation coefficients (i.e. the confusion matrix C) to Aubert's phoneme confusion matrix. For the 194 cases where the hand generated matrix contained non-zero values, the correlation was $\rho = 0.365$. Using the entire confusion matrix except for the identity cases, the correlation coefficient was $\rho = 0.285$. In both cases, there

is a statistically significant (P >0.001) correlation between the hand-generated confusion matrix and the one obtained from the PDP network by the spreading activation model.

4 Word Sense Disambiguation

At a "later" level of cognition, word sense disambiguation models have been proposed to account for lexical access data found in psychological experiments (cf. Swinney, 1979). One such model is a structured connectionist model developed by Gary Cottrell at the University of Rochester (Cottrell, 1985). Cottrell, using weights derived by hand, demonstrated that a structured connectionist network could distinguish both word-sense and case-slot assignments for ambiguous lexical items, in a manner consistent with experimental results. Presented with the sentence "John threw the fight" the system would immediately activate both meanings of "throw," but in a short time would settle in an activation pattern in which a node corresponding with only one of the meanings would remain on. Presented with "John threw the ball" it settle on another meaning. The nodes of Cottrell's network included words (John, Threw, etc.), word senses (John1, Propel, etc.) and case-slots (TAGT (agent of the throw), PAGT (agent of the Propel), etc.).

To duplicate Gary's network via training, we used back propagation to train a network, using a training set in which distributed patterns, very loosely corresponding to a "dictionary" of word encodings[5] were associated with a vector representing each of the individual nodes which would be represented in Cottrell's system, but with no structure. Thus, a typical element in the training set could be, for example, a 16 bit vector (representing a four word sentence, each word as a 4 bit pattern), associated with another 16 bit vector representing the nodes:
Bob1 John1 Propel Threw Fight1 Ball1 Pagt Pobj Tagt Tobj Bob John Threw The Fight Ball

For this example, the system was trained on the encodings of the four sentences:

1. John threw the ball.
2. John threw the fight.
3. Bob threw the ball.
4. Bob threw the fight.

with the output set high for those objects in the second vector which were appropriately associated.

Upon completion of the learning, the activation spreading algorithm was used to derive a table of connectivity weights between the output units. These weights were then transferred into the Rochester Connectionist Simulator (Goddard, et.al., 1987), the same simulation method used by Cottrell, and the activation spreading model was used to examine the results. Using the activation spreading method described in section 2, results similar in time-course and behavior to those produced by Cottrell's model were seen. Thus:

[5]In a realistic , these would be replaced by actual signal processing outputs or other representations of actual word pronunciation forms. This technique of using a random encoding is based on the work of Jeff Elman (1988).

1. Activation from the nodes corresponding to *john, throw, the,* and *fight* cause a positive activation at the node for "Throw" and a negative activation at the node for "Propel."

2. Activation from *john throw the ball* spread positively to "Propel" and not to "throw."

3. Activation at *TAGT* and *TOBJ* spreads positive activation to *Throw* and not to *Propel.*

4. Activation at *PAGT* and *POBJ* causes a spread to *Propel* but not to *Throw.*

(We have also used this approach to test more complex sentences, still within the framework of Cottrell's system. Similar results have consistently been obtained.)

5 Priming Effects

A consistent effect observed in experimentation with humans has been the priming effects that are largely responsible for the belief in an autonomic activation-spreading system[6]. Such effects, however, are not exhibited in even the recurrent PDP models. One particular aspect of these effects is the ability for "semantic" expectations to prime recognition and categorization tasks. For example, when expecting a "vowel," *e* will be more quickly recognized as a letter than if primed to expect a "number." Thus, an activation spreading method should allow prior activation of a concept to facilitate recognition of an example of the concept (vowel/*e*, etc.). This facilitation can appear both in a shortened time course to recognition, or in a preference for a recognition of an ambiguous signal based on an expectation.

We have recently begun experimention which shows that priming effects may be induced via the activation spreading method described in section 2. That is, given a previous activation at a particular node, we may cause some other node to "win" more activation energy, faster, than it would if the previous activation was either missing or was on some other node.

To demonstrate this effect, we trained a 12-4-12 auto-associative network[7] to recognize a training set in which the twelve inputs corresponded to the numbers 0 through 9, and two extra inputs, one of which was on when an odd number was presented, the other on when an even number was presented (we'll call these nodes even and odd for simplicity). Thus, the numeral 3 would be represented as 0's in positions corresponding to other numbers and to even, and 1's in the positions corresponding to the number itself and to odd, that is "0 0 0 1 0 0 0 0 0 0 1 0." After training, on the encodings of all 10 numerals, the covariance coefficients were computed and transferred into the Rochester Connectionist Simulator, as in the previous section.

To test for the ability to simulate priming, ambiguous activation patterns were used to observe network behavior. Thus, the system might have the output corresponding to the numeral 3 activated at a strength of .4, and the numeral 6 represented at a value of .6. As the system settled, one or the other of these nodes would become positive, while the other

[6] A good discussion of these effects and related experiments can be found in Anderson and Bower (1983).

[7] that is, one in which the inputs and outputs in the training set were identical

would become negative. Where no other activation was introduced, the node with higher activation would win out over the other[8].

Priming effects are introduced in these networks by first activating either the node *odd* or the node *even* for a short time, followed by the presentation of the ambiguous input. In this situation, the following sorts of behaviors are seen:

1. Where the direction (odd or even) and the number with the larger activation are the same, that number gains ascendancy (becomes more positive while the other node becomes negative) more quickly than where no prior activation is used.

2. Where the direction and the item with less activation (.4) are different, the item with less activation will end up in ascendancy (as opposed to becoming negative as happens without the presence of the priming activation).

Thus, this activation does correspond well with priming effects. It should be noted that as this technique has only been used on quite small data sets, there is a question as to whether the results will scale for more significant trials. Experimentation in this direction is currently underway.

6 Conclusions

In this paper, we have presented some evidence that the sorts of representations learned, via training, by connectionist networks, may have the necessary properties to be able to demonstrate several effects known to occur in human cognition. Using a spreading activation model over the PDP network, we have shown evidence for the ability to model a simple recognition of perceptually related items (the phoneme confusions) and linking of semantically related items (the lexical items in Cottrell's model). In addition, we've discussed some preliminary evidence that priming effects, a robust phenomena in the activation spreading literature, can be shown in this spreading-activation model. Thus, we have presented evidence demonstrated that distributed representations, such as those learned by these networks, could possibly account for activation spreading effects as is required to account for many known psychological results.

As noted in the introduction, however, the evidence presented in this paper should not be taken as a direct model of human cognitive processing and activation spread. The differences between multiple-layer, back propagation trained, PDP networks and the either the hardware of the human brain or the human cognitive apparatus are many. Thus, the work presented in this paper, and numerous extensions thereto, are *necessary* to demonstrating that a subsymbolic substrate could implement human-like cognition; however it is far from sufficient. We are currently examining the use of this technique in more complex networks that simple feed-forward, completely connected networks, and believe that similar effects must be shown if these networks are to be taken seriously as models underlying human cognition.

[8]We should note, however, that these networks often fail to stabilize, and thus all weights go to 0 after a relatively short time. This is the main reason why we categorize these results as "preliminary."

References

[1] Anderson, J.R. *The Architecture of Cognition,* Harvard University Press, Massachusetts, 1983.

[2] Anderson, J.R. and Bower, G.H. *Human Associative Memory,* Lawrence Erlbaum Associates, New Jersey, 1979.

[3] Aubert, X. (1988). Personal communication.

[4] Bourlard, H., Morgan, N., Wellekens, C.J. (1989c). Statistical Inference in Multilayer Perceptrons and Hidden Markov Models with Applications in Continuous Speech Recognition, to appear in *Neuro Computing, Algorithms, Architectures and Applications*, NATO ASI Series.

[5] Cottrell, G.W. *A Connectionist Approach to Word Sense Disambiguation* Doctoral Dissertation, Computer Science Department, University of Rochester, May, 1985.

[6] Elman, J.L. (1988). Finding Structure in Time, *CRL Tech, Report 8801*, University of California, San Diego,

[7] Feldman, J.A. and Ballard, D.H. Connectionist models and their properties *Cognitive Science,* 6, 1982. *205-254.*

[8] Goddard, N., Lynne, K. and Mintz, T. *The Rochester Connectionist Simulator*, TR233, Dept. of Computer Science, University of Rochester, 1987.

[9] Hendler, J. Marker-passing over microfeatures: Towards a hybrid symbolic/connectionist model *Cognitive Science*, 13(*1*), March, 1989 *p. 79-106.*

[10] Hendler, J.A. *Integrating Marker-passing and Problem Solving: A spreading activation approach to improved choice in planning* Lawrence Erlbaum Associates, N.J., Dec. 1987.

[11] Rumelhart, D.E., McClelland, J.L. and the PDP Research Group *Parallel Distributed Computing, (Volume 1 and Volume 2)* MIT Press, Cambridge, Ma., 1986.

[12] Swinney, D.A. Lexical access during sentence comprehension: (Re)Consideration of context effects *Journal of Verbal Learning and Verbal Behavior,* 18, 1979, *645-659.*

Resolution of Structural Ambiguities in Sentence Comprehension: On-line Analysis of syntactic, lexical, and semantic effects[1]

Gerhard Strube, Barbara Hemforth, and Heike Wrobel
Ruhr University, Bochum, FRG

Temporal onset and relative strength of effects of lexical preference and world knowledge were analyzed in relation to syntactic preferences in a series of experiments on the interpretation of sentences that contained structurally ambiguous prepositional phrases. We found a significant effect of lexical preference (i.e., of main verb subcategorization) that was established before a decisive influence of world knowledge. Since the sentences were in German, transforming them into the "Perfekt" (present perfect) tense allowed for studying the relative effects of those factors in head-final verb phrases. Under these conditions, world knowledge suppressed the influence of lexical preference, which nevertheless affected decision times. In all cases, a slight bias in favor of early closure was found. The results are discussed with respect to contrasting theories of psychological principles in sentence parsing.

To assess how different sources of knowledge, like syntactic, semantic or lexical, and world knowledge, influence the way we process natural language, constitutes one of the main goals for psycholinguistic studies (e.g., Bever, 1970, Crain & Steedman, 1985, Flores d'Arcais, 1982, Ford, 1986, Frazier, 1987, Strube, Hemforth & Wrobel, 1990). We believe that language understanding entails the incremental construction of a mental model (Johnson-Laird, 1983, van Dijk & Kintsch, 1983), making use of those sources of knowledge interactively. Sentences with global structural ambiguities are optimally suited for studies of the relative effects of those sources of knowledge.

The materials used in our experiments consists of sentences of the following kind:

(a) *Susanne verzierte die Torte mit der süßen Sahne.*
(Susan garnished the fancy cake with the sweet cream.)

Here we get two possible readings:

(a1) *instrumental:* *garnished ... with the sweet cream*
(a2) *attributive:* *the fancy cake with the sweet cream*

The structural interpretations (a1) and (a2) differ with respect to the attachment of the prepositional phrase (PP). In (a1), the PP would be attached as an instrument to the verb, in (a2), it would serve as a specification of the second noun phrase (NP "the fancy cake"), and would be integrated into a complex NP.

Three kinds of linguistic hypotheses have been stated regarding the processing of structural ambiguities: (1) purely syntactic principles (Frazier, 1987, Kimball, 1973), (2) lexical preferences (Ford, 1983, 1986, Ford, Kaplan & Bresnan, 1982, Mitchell & Zagar, 1986), (3) hypotheses concerning dominant influences of pragmatic constraints and general world knowledge (Altmann, 1988, Altmann & Steedman, 1988, Crain & Steedman, 1985, Johnson-Laird, 1983).

---

[1] The research reported here was funded by the Deutsche Forschungsgemeinschaft (the German National Association for Scientific Research) under contract no. Str 301/1-1 and Str 301/1-2. We thank Robin Hörnig and Christoph Scheepers for running subjects and assisting in data analysis, Joachim Winzier and Norbert Tkocz, who were responsible for computer control of the experiments, and Waltraud Brennenstuhl-Ballmer for linguistic advice.

A well-known syntactic principle is minimal attachment (Frazier & Fodor, 1978). In essence, it postulates that the reading containing the least number of precategorial nodes in the phrase-structure tree is the preferred one. Other sources of knowledge, i.e., lexical and semantic preferences, come in later, operating as a kind of filter (Frazier, 1987). In contrast, Lexical Functional Grammar calls for direct inclusion of lexical knowledge in the process of sentence parsing and interpretation (Ford, Bresnan & Kaplan, 1982). Based on the LFG framework, although not strictly dependent on it, Ford (1986) developed four parsing guidelines, or syntactic closure principles. The central principle of lexical preference states that whenever the lexicon contains several possible subcategorization frames for the same verb, they will be ordered according to lexical preference, and this order will affect how a sentence is parsed. With respect to our example sentence (a), this amounts to the following prediction: If a three-slot frame (subject, object, oblique instrument) is prioritized for the verb "to garnish", (a1) will be chosen as the preferred reading. If otherwise, i.e., the two-slot frame (subject, object) having priority, parsing of (a) would be subjected to Ford's final argument principle: when the last slot has been filled and the sentence still continues, the object NP is not closed before the sentence has been parsed completely, resulting in the construction of a complex NP, or reading (a2).

The above theories of psychological parsing rely on grammatical knowledge, i.e., on language-specific knowledge as contrasted to general world knowledge (Felix, Kanngießer & Rickheit, 1986). They cannot explain the principles that guide the analysis of a sentence like

(b) Susanne verzierte die Torte mit dem delikaten Mokkageschmack.
(Susan garnished the fancy cake with the delicious mocca flavor.)

Obviously, knowledge is needed about possible or probable pairings between entities and suitable attributes, or between verbs and instruments (or ornatives). In (b), a mocca flavor is hardly a possible instrument for decorating a cake, nor a decoration in itself. Crain and Steedman (1985) express this line of reasoning through their principle of a priori plausibility: Other things being equal, that reading of a sentence will be preferred which appears more plausible in the face of general or domain-specific world knowledge, or the current discourse. This is in turn explained by a principle of parsimony: The more plausible an interpretation, the less presuppositions, or changes to the mental model currently held, are needed to understand the sentence.

In the following, two experiments from a larger series are reported in which we studied the processing of sentences with structurally ambiguous prepositional phrases (all of them using the German preposition "mit", i.e., "with"). Lexical preference and world knowledge were varied in both experiments. Due to a variation of tense (which put the main verb at the end of the sentence according to German grammar), these sources of knowledge could be applied at different times in the two experiments. Experiment 1 used German "Imperfekt", with head-initial position of the verb like (a) or (b), whereas the "Perfekt" sentences of Experiment 2, like (c), are characterized by head-final position of the verb (in the form of the participle). Thus lexical preference comes into play before world knowledge in Experiment 1, but only at the end of the sentence in Experiment 2.

(c) Susanne hat die Torte mit der süßen Sahne verziert.
(Susan has the fancy cake with the sweet cream garnished.)

Experimental design

Lexical preference (L) and world knowledge (W) were included as factors in the design. Lexical preference had the following values:

L1 preferred subcategorization frame with 3 slots (subject, object, oblique instrument or ornative)(3-slot verbs),
L2 preferred subcategorization frame with 2 slots (subject, object)(2-slot verbs),
L3 no clear preference of either the 2-slot or 3-slot frame (neutral verbs).

World knowledge also entered the design in three categories:

W1 preferred verb-modifying (e.g., instrumental) interpretation,
W2 preferred object-specifying interpretation,
W3 ambiguous, both readings possible with respect to general knowledge.

Here are some examples of the resulting combinations:

L1 W1 *Susan garnished the cake with the small icing funnel.*
W2 *Susan garnished the cake with the delicious mocca flavor.*
W3 *Susan garnished the cake with the sweet cream.*

L2 W1 *Laura ordered the document with the short phone call.*
W2 *Laura ordered the document with the black binding.*
W3 *Laura ordered the document with the yellow form.*

L3 W1 *Bill frightened the child with the old ghost story.*
W2 *Bill frightened the child with the little snub-nose.*
W3 *Bill frightened the child with the small toy pistol.*

Construction of materials

In order to construct sentences according to the nine conditions, we had to conduce some extensive pre-tests. Following a linguistically founded preselection of verbs with the help of a pertinent dictionary (Ballmer & Brennenstuhl, 1986), we ascertained the verbs' lexical preference with several sentence completion tasks. For each of the classified verbs, we then built three sentences according to the three world knowledge conditions. This classification was verified with another pre-test in which subjects' comprehension of structurally disambiguated versions of the sentences were recorded.

Experimental Technique

In both experiments, we used the technique of a continuous semantic decision task which we developed in the context of previous project work on the comprehension of temporal clauses. According to this technique, subjects read sentences on a computer screen word by word. By pressing one of two buttons, the subject indicates her comprehension of the sentence and at the same time controls the speed of the presentation. As soon as the button is pressed, the actual word disappears from the computer screen and the next word appears.

This procedure combines a method first used by Aaronson (1976) - the recording of word reading times during RSVP (rapid serial visual presentation) of the sentence material - with the method of continuous reaction which was developed by Wickelgren, Corbett & Dosher (1980) and improved by Schmalhofer (1986).

Experiment 1

Method

Procedure: 21 subjects (students of the Ruhr-University of Bochum with German mother-tongue; no students of psychology) participated in the experiment. Subjects were run individually in half an hour sessions. They were paid for their efforts. The continuous semantic decision task mentioned above was realized in the following way: word by word, subjects had to decide or hypothesize, respectively, whether the prepositional phrase of the actual sentence was more likely to modify the verb (left button), or the object (right button). After the end of the sentence, a question mark appeared on the computer screen. Now the subjects had to indicate their decision again. To make sure that they considered the whole sentence, subjects were asked an additional question about the actual sentence's semantic content. For warming up, seven filler sentences were presented in the beginning of the experiment.

We recorded the following dependent variables:

(a) on-line decisions about the attachment of the prepositional phrase for each word and after the end of the sentence

(b) inspection times for each word and for the decisions after the sentence, i.e the time interval between the appearance of a word on the computer screen and the moment when the button is pressed

Hypotheses

Assuming that both sources of information, world knowledge and lexical knowledge, guide processing immediately, as soon as the corresponding informations are available, the following course of decisions should be found: the verb should deliver the preferred subcategorization frame, so that from the second position onwards significant main effects should be found. For two-slot frames only few decisions should be made in favor of the "verb-modifying" interpretation, whereas this interpretation should be strongly preferred for three-slot frames. No preferences are expected for neutral verbs. In the unambiguous world knowledge conditions decisions should definitely be influenced, as soon as the noun in the prepositional phrase is processed. Since the semantic analysis of some adjectives constrains the choice of following nouns (the girl with the blond ...), at this position (position 7) first effects of world knowledge may be found.

Facilitiation effects should be demonstrable for decisions as well as inspection times, when informations from both sources are congruent, whereas incongruent informations should complicate processing. From earlier experiments with different materials we expect that world knowledge dominates the final interpretation, so that lexical preferences can only influence final interpretations of semantically ambiguous sentences.

Statistical Methods

For both experiments, decisions were statistically analyzed by stepwise adaptations of hierarchical loglinear models with lexical preference and world knowledge as independent variables.
Inspection times were tested by within subject analyses of variance with the same independent variables. The word length, operationalized as the number of letters for each word, was included as covariate, if needed.

Results

Decisions:

As shown in figure 1, there is a significant main effect of lexical preference from position 4 to position 7 (from the noun of the simple object NP to the adjective in the PP). The significant effects are due to the fact that at these positions sentences are judged more often in favor of a verb-modifying interpretation, if the sentence contains a three-slot verb. Neutral and two-slot verbs do not differ significantly. There is a weak tendency ($p < .11$) for the main effect at position 3 (the article of the simple object NP).
Contrary to our hypothesis, the main effect does not show up at position 2 and can no longer be found at the end of the sentence even for semantically ambiguous sentences.

As expected, sentences were judged according to the world knowledge conditions from the adjective of the PP onwards (see figure 2). In semantically ambiguous sentences PPs were more often interpreted as verb-modifying (60.5%).

Inspection times

No significant differences between the three lexical conditions were found. At the end of the sentence (position 8), there is a significant main effect due to world knowledge. The unambiguous conditions are processed faster (verb-mod. vs ambig.: $F=3.95$, $df=1,19$, $p<.07$; obj.-mod. vs ambig.: $F=21.03$, $df=1,19$, $p<.001$).

Figure 1.

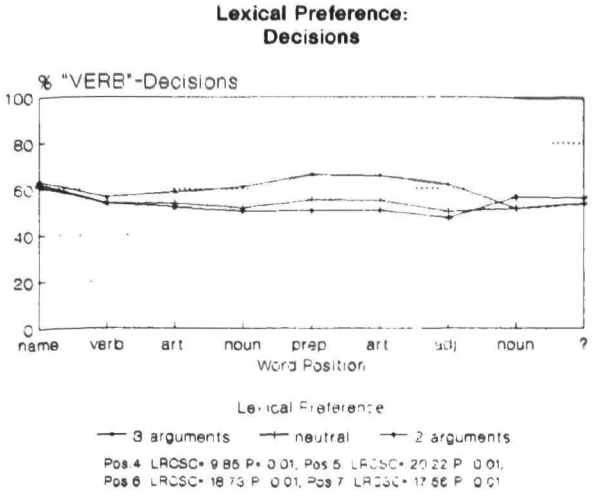
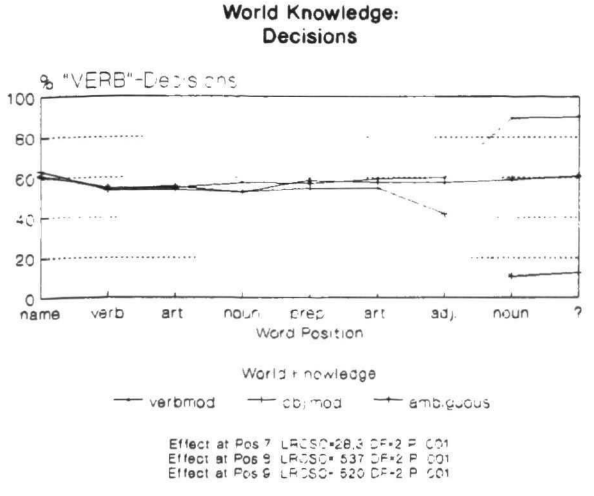


Figure 2.



Congruent informations from both sources (world knowledge and lexical preference) lead to significantly shorter inspection times than incongruent informations (verb-mod./verb.-mod.: mean= 2729 msec, obj.-mod./obj.-mod.: mean=2525 msec, verb.-mod./obj.-mod.: mean=3638 msec, obj.-mod./verb-mod.: mean=3375 msec; $F= 4.43;DF=2,40;p < .02$).

Experiment 2

Method

Because of the head-final position of the main verb in German perfect tense sentences the availability of world knowledge and lexical preference is reversed. For our second experiment only the tense of the sentences was changed from "Imperfekt" (past tense) to "Perfekt" (present perfect). Method and experimental design, and consequently dependent and independent variables were identical to those of Experiment 1.

Hypotheses

As in the first experiment we expect that world knowledge is used for disambiguation immediately, i.e. as soon as the corresponding informations are available. According to this hypothesis the course of decisions should develop in the following way: When the adjective and the noun of the prepositional phrase are processed there is enough information to decide whether the prepositional phrase can be attached to the simple object NP. If attachment is possible subjects should prefer the object modifying interpretation, otherwise they should decide that the prepositional phrase will modify the verb. The attachment to the simple object NP implies that the new information (PP) can be combined with an already established entity in the mental model of the sentence. This integration should be less effortful, i.e. less time-consuming than the establishment of a new entity in the verb modifying case. If world knowledge supports the verb modifying interpretation the prepositional phrase delivers information about possible verbs. So the main verb should be processed faster in this condition.

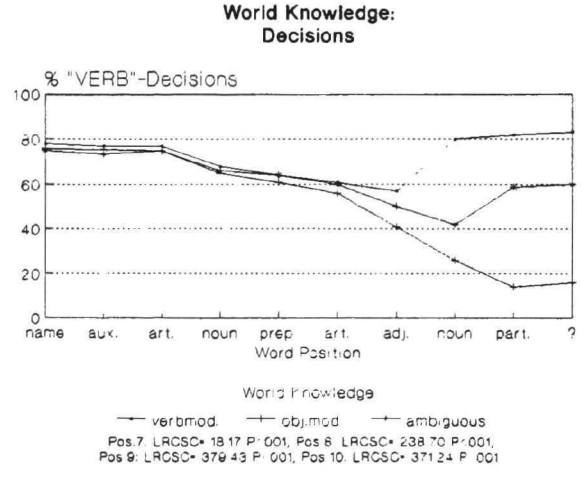
In our first experiment we did not find any effects of lexical preference on decisions as soon as world knowledge information was fully available. Because of the head-final position of the verb we do not expect any effects of lexical preference in this experiment. The interaction of world knowledge and lexical preference concerning inspection times should still be demonstrable.

Results

Decisions

No effects of lecical preference were found. The expected main effect of world knowledge is already to be found at the seventh position (adjective) and stays significant up to the decision after the end of the sentence. Decisions were made according to the world knowledge conditions (see figure 3).

Figure 3.

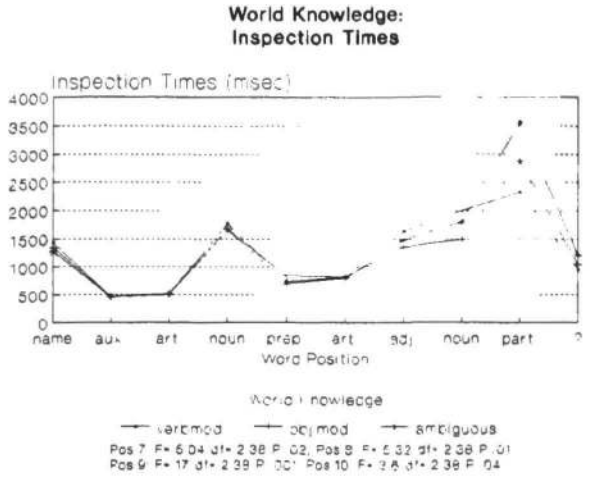


Inspection times

There were no significant effects of lexical preference. As shown in figure 4, a significant main effect of world knowledge is present from position 7 (adjective of the PP) on. As expected, inspection times for positions 7 and 8 (adjective and noun of the PP) were shorter

if world knowledge supported the object-modifying interpretation (position 7: F=6.00, df=1,19, p < .03, position 8: F=3.74, df=1,19, p < .07). At position 9 (main verb participle) the order of means is reversed (F=12.6, df=1,19, p < .01), again as expected. This results in a significant interaction between world knowledge and position (F=12.88, df=2,37, p < .001). Unambiguous world knowledge leads to significantly shorter inspection times at the end of the sentence (position 9; verb-mod. vs amb.: F=19.14, df=1,19, p < .001; obj.-mod. vs amb.: F=14.02, df=1,19, p < .01).

Figure 4.



As in Experiment 1, a significant interaction between lexical preference and world knowledge was found at the end of the sentence(F=4.83, df=2,37, p < .02). The effect is confined to 3-slot verbs where congruent informations from both sources lead to shorter inspection times than incongruent informations (verb-mod.: mean=2107 msec; obj.-mod.: mean =3167 msec; F=14.37, df=1,20, p < .01).

Discussion

Summing up, the results of our experiments show that both sources of knowledge are drawn upon during sentence processing, and that they are used on-line, i.e., human parsing proceeds incrementally. The various sources of knowledge are, however, not equally important in determining the interpretation of a sentence.

(1) An effect of lexical preference could be demonstrated in Experiment 1, although it did not quite conform to expectations. For one, availability of lexical information (contingent on the verb) does not show an effect immediately, but only builds up a significant difference from the noun of the object NP onward. This testifies against immediate **full** processing, suggesting gradual processing of lexical information. Most important, lexical effects are present before the PP is processed. Therefore we conclude that lexical preference operates as a source of information guiding further syntactic analysis (lexical guidance, Ford, 1986, Holmes, 1987) rather than a filter applied to the result of autonomous syntactic analysis (Frazier, 1987).

Verb frames for verbs with preferred 2-slot readings (without an oblique instrument) and for verbs without a clear preference do not give rise to statistically significant differences during sentence processing. In other words, preferredly 2-slot subcategorization frames do not generate a specific expectation. Even at the end of a sentence, no effect shows up, and world knowledge alone

determines the interpretation.

(2) World knowledge comes into play as soon as possible, i.e., when reading the adjective of the PP. Inspection times show that unambiguous world knowledge facilitates processing.

In head-final sentences (German present perfect) integrating the PP is easier if it can be integrated with the object into a complex NP. On the other hand, the predictive value of the instrument/ornative noun of the PP facilitates processing of the main verb participle.

(3) For both experiments, congruent information at the end of the sentence (i.e., both world knowledge and lexical preference in concordance) leads to shorter processing times. (In Experiment 2, this effect is confined to 3-slot verb frames.)

(4) World knowledge and lexical preference show two major differences. While world knowledge dominates the interpretation and is processed immediately, the effects of lexical preference take some time to build up, are generally much weaker, and disappear toward the end of a sentence.

References

Aaronson, D. (1976). Performance theories for sentence coding: Some qualitative observations. *Journal of Experimental Psychology: Human Perception and Performance*, *2*, 42-55.

Altmann, G. (1988). Ambiguity, Parsing Strategies, and Computational Models. *Language and Cognitive Processes*, *3*, 73-97.

Altman, G. & Steedman, M. (1988). Interaction with context during human sentence processing. *Cognition, 30.*

Ballmer, T. T., & Brennenstuhl, W. (1986). *Deutsche Verben*. Tübingen: Gunter Narr Verlag.

Bever, T. G. (1970). The cognitive basis for linguistic structures. In J. R. Hayes (Ed.), *Cognition and the development of language* (pp. 279-362). New York/London/Sydney/Toronto: John Wiley & Sons, Inc.

Crain, S., & Steedman, M. (1985). On not being led up the garden path: The use of context by the psychological syntax processor. In D. R. Dowty, L. Karttunen, & A. R. Zwicky (Eds.), *Natural language parsing* (pp. 320-358). Cambridge: Cambridge University Press.

Felix, S., Kanngießer, S. & Rickheit, G. (1986). Antrag auf Einrichtung eines Schwerpunktprogramms "Kognitive Linguistik". *Deutsche Forschungsgemeinschaft, Bonn*.

Flores d'Arcais, G. B. (1982). Automatic syntactic computation and use of semantic information during sentence comprehension. *Psychological Research*, *44*, 231-242.

Ford, M. (1983). A method for obtaining measures of local parsing complexity throughout sentences. *Journal of Verbal Learning and Verbal Behavior*, *22*, 203-218.

Ford, M. (1986). A computational model of human parsing processes. In N. E. Sharkey (Ed.), *Advances in cognitive science* (Vol. 1, pp. 259-275). Chichester, England: Ellis Horwood (Wiley).

Ford, M., Bresnan, J., & Kaplan, R. M. (1982). A competence-based theory of syntactic closure. In J. Bresnan (Ed.), *The mental representation of grammatical relations* (pp. 727-796). Cambridge, MA: MIT Press.

Frazier, L. (1987). Sentence processing: a tutorial review. In M. Coltheart (Ed.), *The psychology of reading* (pp. 559-586). Hove/London/Hillsdale: Lawrence Erlbaum.

Frazier, L., & Fodor, J. D. (1978). The sausage machine: A new two-stage parsing model. *Cognition*, *6*, 291-325.

Holmes, V.M. (1987). Syntactic parsing: in search of the garden path. In M. Coltheart (Ed.), *The psychology of reading* (pp. 587-599). Hove/London/Hillsdale: Lawrence Erlbaum.

Johnson-Laird, P. N. (1983). *Mental models. Towards a cognitive science of language, inference, and consciousness*. Cambridge, MA: Harvard University Press.

Kimball, J. (1973). Seven principles of surface structure parsing in natural language. *Cognition*, *2*, 15-47.

Mitchell, D. C., & Zagar, D. (1986). Psycholinguistic work on parsing with lexical functional grammars. In N. E. Sharkey (Ed.), *Advances in cognitive science* (Vol. I, pp. 276-289). New York: John Wiley.

Schmalhofer, F., & Glavanov, D. (1986). Three components of understanding a programmer's manual: Verbatim, propositional, and situational representations. *Journal of Memory and Language*, *25*, 279-294.

Strube, G., Hemforth, B. & Wrobel, H. (1990). Auf dem Weg zu psychologisch fundierten Modellen menschlicher Sprachverarbeitung. In: S. Felix, S. Kanngießer & G. Rickheit (Eds.), *Kognitive Linguistik*. Opladen: Westdeutscher Verlag.

Prof. Dr. Gerhard Strube
Ruhr-Universität Bochum, Fakultät für Psychologie
Postfach 10 21 48
D-4630 Bochum 1
Federal Republic of Germany

A connectionist model of attentional enhancement and signal buffering.

Judith M. Shedden and Walter Schneider
University of Pittsburgh

Abstract. The connectionist/control simulation of attentional enhancement, signal maintenance, and buffering of information is described. The system implements a hybrid connectionist architecture incorporating auto-association in the hidden layer and gain control on the hidden and output layer. The structure of the model parallels major features of modular cortical structure. The attentional selection simulations show that as one channel is attenuated, the system exhibits attentional capture in which only the more intense stimulus is transmitted to higher levels. The signal maintenance simulations show that small levels of auto-associative feedback can faithfully maintain short bursts of input for extended periods of time. With high auto-associative feedback, one module can buffer information from a previous transmission while the module blocks the interference resulting from concurrent transmissions. The combination of auto-associative feedback and gain control allow extensive control of information flow in a modular connectionist architecture.

This paper examines signal control issues in a connectionist processing system. It examines three basic cognitive operations of **attentional enhancement**, **signal maintenance**, and **buffering** of information. In human processing these operations represent control functions that are often associated with mechanisms of attention. Attention is the selection for processing of some subset of available information in the environment. For example, when humans switch attention from one item to another there is an attentional capture effect where they perceive one signal to the almost total exclusion of another signal (e.g., Treisman & Riley, 1969). Information that is briefly presented is maintained in a short-term sensory buffer that is available for a substantial period after the sensory stimuli are removed (e.g., Sperling 1960). Humans can also buffer information in short-term memory while encoding and acting on new information (Klapp, Marshburn, & Lester, 1983).

Traditional Models of Attention. The mechanism of attentional focus has been a topic of interest to psychologists for years (see Shiffrin, 1988). Broadbent's (1957, 1958) theory of selective attention suggests a filter mechanism, in which information from all channels is initially processed in parallel, but at some point in the system information converges on a limited capacity channel. The model originally stated that selection is all-or-none, with no information being passed from other unattended channels. This view was modified when experiments showed that unattended information is available under some circumstances. Treisman's (1960) attenuation model is a modification of the filter model. It states that both the attended and unattended channels receive processing, but the processing in the attended channel is complete, and the processing in the unattended channel is 'attenuated' to some degree. Later models have combined selective processing with limited parallel processing. The Shiffrin and Schneider (1977) model assumes there is a parallel processing of multiple channels of consistent well learned information, and serial controlled processing of novel or inconsistent information. All of these models assume some mechanism of attentional enhancement. From this perspective, the mechanism for enhancement must still be determined.

Connectionist Models of Attention. Recently there have been several connectionist models of attention. The present models simulate either the selection issue or enhancement issue. The Koch and Ullman (1985) model involves a winner-take-all hierarchical model to indicate which pathway is to be selected. The Mozer (1988, and Mozer & Behrmann, 1989) model involves a topographic multidimensional map, with attention represented as a set of units that gate the flow of activity from lower levels in proportion to the strength of the attentional map. The Cohen, Dunbar and McClelland (in press) model examines the effect of the enhancement of the attended channel via control of the resting levels of the attended and unattended pathways. The model is applied to performance in the Stroop task. Our current simulation examines the enhancement effect under the assumption that attention involves changing control parameters of a connectionist module. The selection issue is not dealt

with in this paper and will be addressed in future papers.

Physiology of Attention. The idea of attenuation, or inhibition of an unattended, competing signal is supported by physiological findings. Moran and Desimone (1985) measured the effects of attention on single cells in primate extrastriate visual cortex (area V4). Attention influences the output rate of V4 neurons as measured with microelectrode recordings. The post-stimulus time histograms show a strong attentional effect (Moran & Desimone,1985 and personal communications Desimone). The effect of selective attention is the attenuation of the signal from the unattended stimuli, and not an enhancement of the signal from the attended stimuli. This attenuation is a lateral inhibition within the receptive field of the responding cell, and the reduction in response of the unattended cells is to 1/3 of the attended state. Thus, there is roughly a 3:1 ratio of attended/unattended signal. This ratio is an important one from a modeling standpoint, because it provides a physiological bench mark against which to test the performance of the model.

Cortical neuroanatomy provides connection patterns and unit types that can be a basis for models of attentional functioning. Most of cortex post the first sensory areas (e.g., post V1 in vision) shows a similar layering of cells and cell types. Studies of V2 cortex identify cortical connections and components (Lund, Hendrickson, Ogren, & Tobi, 1981). Cortical processing appears to occur in a modular structure of columns (Mountcastle 1979). The forward information flow passes through two layers of pyramidal cells and inputs into layer 4, then to layer 2-3 pyramidal cells, and then out to the next module (see Lund et. al. 1981). The layer 4 cells connect recurrently to themselves providing a feedback path. In addition there are inhibitory interneurons that are primarily within a layer. There is a special class of axon-axon inhibitory cells called chandelier cells that attenuate large numbers of output cells (see Peters, 1984). This cell can inhibit the output of sets of pyramidal cells. The fast large scale inhibitory effects of chandelier cells make them a good candidate for the attenuation effects seen by Moran and Desimone (1985).

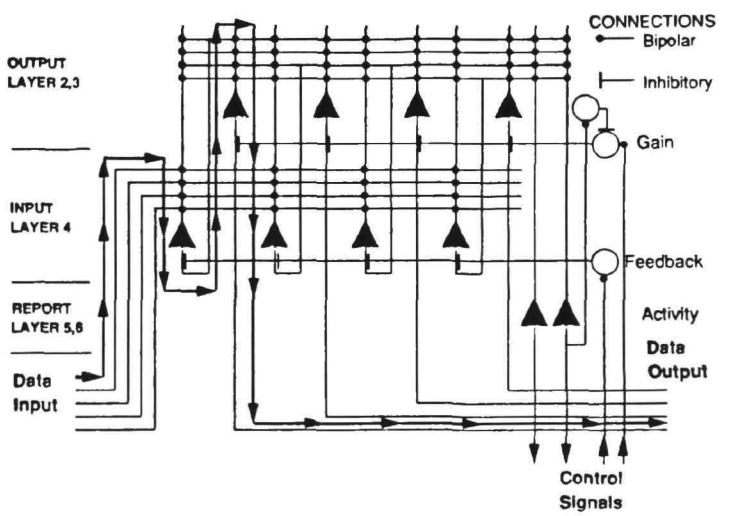


FIGURE 1: A connectionist model of cortical processing. It parallels the cortical processing in that it is a two layer structure (input layer 4 and output layer 2,3). Input feedback and output are controlled by two inhibitory modulation units. Control signals are provided by report units (layer 5,6) transmitting information regarding the activity and priority of the input to more central control structures and influencing the local gating of layer 2,3 activity. These control structures are assumed to influence the feedback and gain signals within a module.

CAP2 Architecture. The CAP2 architecture is a computer simulation environment designed to implement a modular connectionist architecture incorporating the major features of cortical processing to predict human attentional effects. The model consists of a variable number of modules, units, layers, and control elements, which can be combined to create several different connectionist architectural configurations. A module

consists of an input layer of units, an associative matrix, and an output layer of units (see Figure 1). Modules can be added to the system in breadth, so that several input modules connect to one output module. In addition, modules can be added to the system in depth, creating several hierarchical layers.

Control elements are an important part of the CAP2 environment, and include internal feedback control, output gain control, and module activity and priority reports (see Schneider & Detweiler, 1987, for a discussion of the activity and priority report). These control elements are scaler values derived from the activity of the module itself, or assigned from outside the module, and act to modulate attentional processing. Feedback control increases or decreases the strength of the signal already in the system. Gain control increases or decreases the strength of the output signal.

net input = (feedback * internal input) + (gain * external input)

The model includes standard connectionist layers with additional association and control effects. The simulation described in this paper consists of two modules connected hierarchically. Each module is a connectionist network consisting of three layers, including the **data input**, the module **input layer** (standard hidden layer in back propagation), and the **output layer**. Figure 1 shows the layers and connections. Each layer has 50 units, and each unit in one layer is connected by a set of weights to every unit in the adjacent layers. The modules can be cascaded so the output layer of one module is the data input of the next. Learning of input output patterns is accomplished via back propagation (see Rumelhart, Hinton, & Williams, 1986).

The system differs from a standard three layer network in that the hidden layer is connected through an auto-associative matrix to itself (see connections for Layer 4 cells to themselves as well as layer 2-3 cells). The auto-associative matrix is taught to reproduce the hidden layer using delta-rule learning (see McClelland & Rumelhart, 1988).

The system includes control elements that modulate the output from one layer to the next. These control elements are connected in a manner similar to the connections of cortical chandelier cells (e.g., one unit providing a scaler reduction of the population of units to which it is connected). The two control units determine **feedback** and **gain** (right of Figure 1). Manipulation of the feedback control element affects the strength of the signal through the auto-association matrix. This allows the system to hold a signal after the external input is turned off. The signal can be perpetuated by cycling through the auto-associative matrix. Control of the gain cell limits the output of the information from one module to the input of the next. The model also includes report cells used for determining where to switch attention and automatic processing within a module (see Schneider 1985, Schneider & Detweiler 1987). These however are not used in the current model.

Learning. The vector space for the simulation consists of ten input vectors and ten target vectors. Activation levels have maximum values of 1.0, minimum values of -1.0, and a resting activation of 0.0. Input vectors are of length 50, with initial activations set randomly to 1.0 or -1.0, and then clipped to 0.9 or -0.9 respectively for each unit. Target vectors are also of length 50, with initial activations set randomly to 1.0 or -1.0 for each unit. The set of all input and target vectors are forced to have correlations below 0.15 with all other vectors in the set.

Input vectors and target vectors are paired, and training of the network consists of back propagation of error (Rumelhart, et. al. 1986) after presentation of each input pattern. The activation function on the output is a logistic:

$$\text{activation} = \frac{1}{1 + e^{-\text{netInput}}}$$

The weights of the autoassociation matrix are changed using delta rule learning (see McClelland & Rumelhart, 1988), with the hidden layer vector as the input and the target. One epoch is defined as the presentation of each input pattern once, although the presentation order is randomized for each epoch. The network is trained for 15 epochs, at which point the system has learned to a criterion of 100% accuracy and correlations between output and target vectors are above 0.98 for each input pattern.

The network learned 10, 50, and 100 vector pairs of 50 units each in 2, 7 and 13 epochs, perfectly choosing the best matching response for the input. The correlation between output and target vectors climbs more slowly reaching correlations of .98, .73, and .55 respectively. This performance illustrates the large potential storage capacity of the model, which is important when relating model performance to brain performance. One cortical hypercolumn has been estimated to contain tens of thousands of cells (Mountcastle, 1979), suggesting that information is coded not as a single unit being on, but rather vectors in which a subset of units are turned on. We have worked with 10, 25, 50 and 200 unit vectors and find the performance of the model works well with large vectors. We use 50 element vectors that allow coding of large numbers of vectors with most vectors showing little correlation. The 50 element vectors do not require the long simulation times of very large vectors. All further testing discussed below is done with input and target vector sets of ten vectors each, after learning is completed.

Attentional Enhancement and Attentional Capture. A basic characteristic of attentional switching in humans is the attentional capture effect. As one moves attention from one stimulus to another, there is a sudden change from perception of the first stimulus to perception of the second. One does not see a mixture of the two stimuli. This is illustrated by observing a Necker cube. One perceives it in one orientation or the other, and this perception may switch between orientations, but one does not see a gradual shift or a combination of the two orientations. This effect is one of the characteristics investigated in this model.

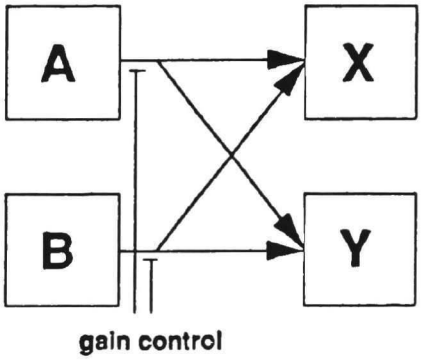


FIGURE 2: Diagram of attentional control between modules. The connections between modules represent vectors. The gain control is a scaler multiplication of all the elements of the vector. Each module connects to and receives from multiple modules. Attentional enhancement of A involves having a higher gain on the A module relative to the B module. Buffering involves loading a message from A to X and storing the message in X while a second message is sent from B to Y. Since B is connected to both X and Y the X module must hold the A message during the transmission from B (see text).

This simulation examines what is perceived when the relative signal strength of the attended and unattended message is altered. Does the receiving module get a clear signal of one of the messages? At what relative strength are the messages clear? Figure 2 illustrates the basic structure of the model. The A and B signals are input to the X module. As the relative strength of the two inputs changes, the X module settles on either the A or B signal. The two input vectors are presented to the network simultaneously for eleven testing trials. On each trial the gain (strength) of each vector is increased or decreased by 0.1, so that the total strength of both vectors is always 1.0. This is done for all combinations of two input vectors. Note the attentional capture tests were done with learning turned off, using the auto-associative and normal associative matrices developed in learning the 10 vector patterns.

The relative gains were:

Trial:	1	2	3	4	5	6	7	8	9	10	11
Vec A Gain:	1.	.9	.8	.7	.6	.5	.4	.3	.2	.1	.0
Vec B Gain:	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	1.

The question being asked is what vector or combination of vectors will be received by the output module? Accuracy is determined as the best match between output and target. Thus, if testing vector A, the trial is correct if the output has a higher correlation with target A than with any of the other possible target vectors. The left panel of Figure 3 shows the accuracy measure and correlations for the averaged data over all input pairs, and shows that at a .7:.3 ratio, all vectors are accurately identified without contamination from the interfering vector. Of more interest is an analysis of individual vector pairs, an example of which is shown in the right panel of Figure 3. Note that with a 0.1 change in relative strength there is a complete shift from perfect recognition of the A vector to perfect recognition of the B vector. In every case a difference of 0.1 in vector activation is enough to cause a sharp transition between the perception of vector A or vector B. There is variability between vector pairs for the position of this transition point, and that accounts for the wider crossover in the averaged data (Figure 3, left panel).

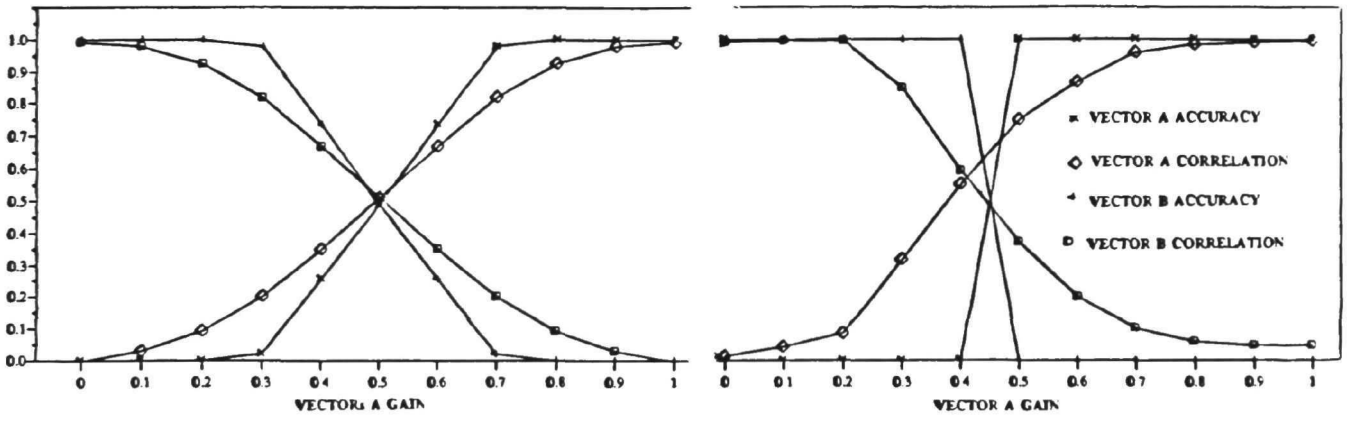


FIGURE 3: Attentional Capture Effects. The left panel shows the averaged data for 45 vector pairs, each pair presented simultaneously to the network. Total gain is always equal to 1.0. For example, if the gain of vectorA = 0.7, then the gain of vectorB = 0.3. Shown are the best match accuracy data and the correlation data. The right panel shows the data for a single vector pair. VectorA and vectorB are presented simultaneously. Best match accuracy and correlation with matching target vector are shown for each vector at each level of gain.

The simulation shows a clear signal capture effect with only the stronger vector being perceived at a strength ratio of 7 to 3. Recall that the attentional data from Moran and Desimone (1985) shows a required difference of 3:1 for the attended/unattended ratio, a point at which the current network provides clear capture. Note in this model that although the attention effect involves gradual attenuation, the network interactions produce an all or none switch to the enhanced signal. All of these tests were with vectors with low correlations; we expect to see some mixture effects with more highly correlated vectors.

Signal Maintenance and Normalization. Much of perception involves the input of brief bursts of information that can then be read out over a period of time. A brief visual stimulus can be read out of iconic storage for over a half a second (e.g., Sperling, 1960). One can fixate a stimulus briefly and then move one's eyes and still recall the initial stimulus. The auto-associative feedback common in cortex provides a mechanism for maintaining short duration signals. In this simulation this is implemented in the layer 4 auto-associative

connections. Feedback in this system provides a way to latch and hold a signal for some extended period of time after the external input has been turned off. If the stimulus burst is very short, feedback can act to boost the signal and thereby normalize the input.

Input vectors are presented to the network for 5 iterations, and then the stimulus is turned off for another 5 iterations. An iteration consists of one pass through the system, from presentation of the input vector to production of the output vector. Five different levels of feedback are tested. The left panel of Figure 4 shows the activity of the hidden layer of units, and illustrates that the signal drops out quickly with a feedback value of 0.0 (maximum possible activity equals 1.0). As feedback increases, activity also increases. At a feedback value of 0.1 the module holds activity stable, accurately maintaining the vector activity pattern in the module. At feedback levels higher than 0.1 the activity continues to increase, but the signal evoked begins to stray from the desired target, and the system begins to 'hallucinate,' or identify stimuli which are not there.

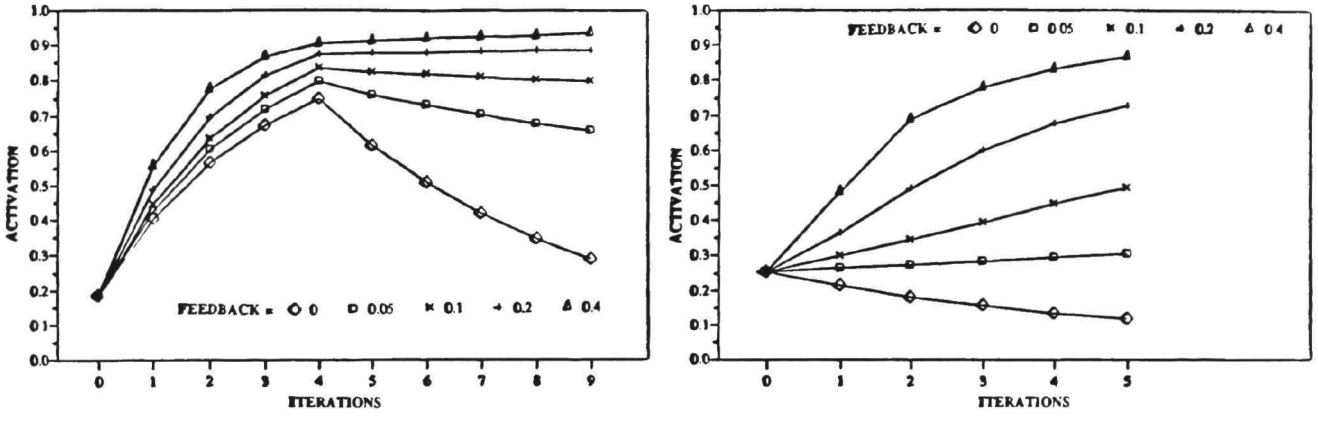


FIGURE 4: Latching and holding a signal. Left panel data are averaged over all stimulus vectors. A stimulus vector is presented for 5 iterations and then removed. Activation levels of the hidden layer are shown at different levels of feedback. At feedback of 0.0 the signal drops out quickly. At feedback of 0.1 the signal holds. At higher levels of feedback activity increases, but the accuracy of the signal begins to deteriorate. Right panel illustrates stimuli presented for only one iteration. Activation levels of the hidden layer are shown for different levels of feedback. At feedback of 0.0 the signal does not hold, but at higher levels of feedback activity increases, approaching normal levels by 4 or 5 iterations.

Feedback also allows the network to latch and increase the activity of a short burst stimulus. Input vectors are presented to the system for 1, 2, 3, or 4 iterations, and then the stimulus is turned off for 5 iterations. As can be seen in the right panel of Figure 4 which illustrates a single iteration stimulus burst, without feedback the signal dies quickly away. At higher levels of feedback the signal is latched and the activity level increases, approaching the strength of a normal signal after 4 or 5 iterations.

Signal Buffering during concurrent loading. When attention is directed to one location, the information can be buffered such that it is not destroyed by attention moving to another location. In reading for example, fixating one word does not clear memory of all previous words. The problem of buffering with concurrent loading is illustrated in Figure 2. Assume the A and B modules both project to the X and Y modules. If the A signal is transmitted it goes to both the X and Y modules. Physiologically, once a signal exits a cortical module it outputs to all sites it is connected to. To buffer information one would like to transmit the A signal to X and the B signal to Y. The problem is that transmitting the B signal to Y produces interference in the X module causing possible loss of information. The same problem occurs when serially loading a set of inputs that must be examined in parallel. In this case the A module must output to the X module for the first stimulus and the Y module for the second. Since the A module is connected to both modules, it is critical that the second transmission to Y does not delete the previous signal to X.

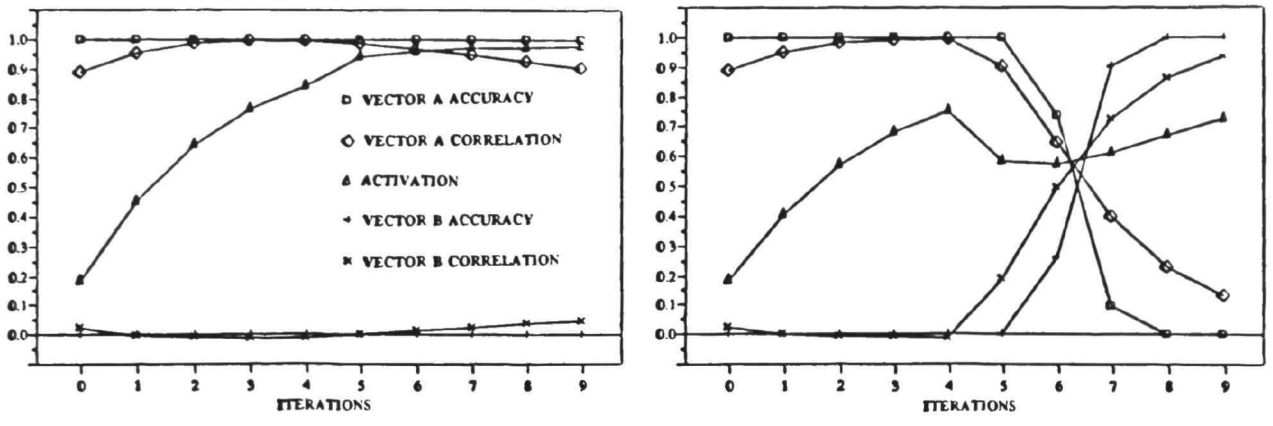


FIGURE 5: Signal maintenance with feedback. In left panel, on iterations 0 to 4 vectorA only is presented. On iteration 5 feedback is increased from 0.1 to 1.5. On iterations 5 to 9 vectorB only is presented. Shown are best match accuracy for vectorA and vectorB, correlation with targets for vectorA and vectorB, and activity of the hidden layer. The signal for vectorA holds throughout. In right panel there is no feedback. The signal for vectorA dies out as soon as the incoming stimulus is replaced with vectorB, and the signal for vectorB is increased.

The auto-associative feedback shown in this model provides a possible mechanism for signal buffering during concurrent loading. If the feedback is high enough, the module will maintain its signal and block out competing signals. In the simulation, a stimulus vector (vectorA) is presented to the network for 5 iterations, allowing the strength of the vectorA signal to build within the module. On the sixth iteration feedback is turned up from 0.1 to 1.5, and a different stimulus vector (vectorB) is presented for 5 iterations. The increase in feedback is sufficient to maintain the signal of vectorA without allowing any contamination from vectorB. The left panel of Figure 5 shows the accuracy and correlations for both stimuli throughout the ten iterations. VectorA remains the strong and accurate signal throughout, and the incoming vectorB does not have an effect. Compare these results to the case (Figure 5, right panel) where feedback was not utilized (feedback = 0.0). In the case where there is no feedback the vectorA signal begins to die as soon as presentation of vectorA is replaced with presentation of vectorB. The competing vectorB signal builds quickly and takes over the system.

Conclusion. The present connectionist control architecture implements some major features of cortical structure. The control of the gain of the output of a module allows selective enhancement of one message by attenuating competing messages. The attenuation of the unattended signal at levels representative of physiological attenuation produces a strong attentional capture effect. The use of auto-associative feedback provides a way to control input at the receiving module. The auto-associative feedback allows the system to maintain briefly presented information for later output after the stimulus has ceased. The feedback control can also latch a signal within a module and reduce the interference of concurrent transmissions that are directed at other modules but still input to that module. This architecture will be explored further to directly simulate human attentional effects, and determine the computational performance of incorporating modern cortical connectivity in a modular connectionist architecture.

References

Broadbent, D.E. (1957). A mechanical model for human attention and immediate memory. Psychological Review, 64, 205-215.

Broadbent, D.E. (1958). Perception and communication. London: Pergamon Press.

Cohen, J. D., Dunbar, K., & McClelland, J. L. (in press). On the control of automatic processing: A parallel distributed processing account of the Stroop effect. Psychological Review.

Klapp, S. T., Marshburn, E. A., & Lester, P. T. (1983). Short-term memory does not involve the "working memory" of information processing: The demise of a common assumption. Journal of Experimental Psychology: General, 112, 240-264.

Koch, C., & Ullman, S. (1985). Shifts in selective visual attention: Towards the underlying neural circuitry. Human Neurobiology, 4, 219-227.

Lund, J. S., Hendrickson, A. E., Ogren, M. P., & Tobin, A. E. (1981). Anatomical organization of primate visual cortex area VII. Journal of Comparative Neurology, 202, 19-45.

McClelland, J. L., & Rumelhart, D. E. (1988). (Eds.), Explorations in parallel distributed processing: A handbook of models, programs, and exercises. Cambridge, MA: MIT Press.

Moran, J., & Desimone, R. (1985). Selective attention gates visual processing in the extrastriate cortex. Science, 229, 782-784.

Mountcastle, V.B. (1979). An organizing principle for cerebral cortical function: The unit module and the distributed system. In F.O. Schmitt & F.G. Worden (Eds.), The neurosciences: Fourth Study Program (pp.21-42). Cambridge, MA: MIT Press.

Mozer, M. C. (1988). A connectionist model of selection attention in visual perception. In Proceedings of the Tenth Annual Conference of the Cognitive Science Society (pp. 195-201). Hillsdale, NJ: Erlbaum.

Mozer, M. C. & Behrmann, M. (1989). On the interaction of selective attention and lexical knowledge: A connectionist account of neglect dyslexia, Tech. Rep. CU-CS-441-89. University of Colorado at Department of Computer Science, Boulder, CO.

Peters, A. (1984). Chandelier cells. In A. Peters & E. G. Jones (Eds.), Cerebral cortex: Vol. 1: Cellular components of the cerebral cortex (pp. 361-381). New York: Plenum.

Rumelhart, D.E., Hinton, G.E., & Williams, R.J. (1986). Learning internal representations by backpropagating errors. Nature, 323, 533-536.

Schneider, W. (1985). Toward a model of attention and the development of automaticity. In M. Posner & O. S. Marin (Eds.), Attention and performance XI (pp. 475-492). Hillsdale, NJ: Erlbaum.

Schneider, W. & Detweiler, M. (1987). A connectionist/control architecture for working memory. In G. H. Bower (Ed.), The psychology of learning and motivation, Volume 21 (pp. 54-119). New York: Academic Press.

Shiffrin, R. M. (1988). Attention. In R. C. Atkinson, R. J. Herrnstein, G. Lindzey & R. D. Luce (Eds.), Steven's handbook of experimental psychology, Volume 2: Learning and cognition (739-809). New York: John Wiley & Sons.

Shiffrin, R. M., & Schneider, W. (1977). Controlled and automatic human information processing. II: Perceptual learning, automatic attending, and a general theory. Psychological Review, 84, 127-190.

Sperling, G. (1960). The information available in brief visual presentations. Psychological Monographs, 74 (Whole No. 498).

Treisman, A. (1960). Contextual cues in selective listening. Quarterly Journal of Experimental Psychology, 12, 242-248.

Treisman, A. M., & Riley, J. G. A. (1969). Is selective attention selective perception or selective response? A further test. Journal of Experimental Psychology, 79, 27-34.

Acknowledgements: This research was supported in part by ONR contract N00014-87K-0397 and N00014-86-K-0678 and ARI contract MDA903-89-K-0174.

Visual Search as Constraint Propagation

Peter A. Sandon
Berrin A. Yanikoglu

Computer Science Program
Dartmouth College

Abstract

A handful of prominent theories have been proposed to explain a large quantity of experimental data on visual attention. We are developing a connectionist network model of visual attention which provides an alternative theory of attention based on computational principles. In this paper, we describe aspects of the model relevant to the dependence of visual search times on display size (number of objects in the stimulus image). Duncan's stimulus similarity theory provides the characterization of the experimental data which we use in simulating and evaluating our model. The characteristics of the network model that support the continuously varying dependence of search time on display size are the constraint propagation search implemented by a winner-take-all mechanism in the attention layer, and the lateral inhibition network within each primitive feature map, which provides the feature contrast needed to filter out background textures. We report the results of simulations of the model, which agree with experimental data on visual attention in human subjects.

Introduction

Visual attention refers to the phenomenon of perceiving relevant parts of a visual stimulus, while ignoring irrelevant parts. Treisman's feature integration theory [Treisman & Gelade 1980] provides one explanation of attentional phenomena in terms of primitive feature maps and how they are combined in recognizing complex patterns. Duncan's stimulus similarity theory [Duncan & Humphreys 1989] is an alternative explanation which is based on two similarity measures applied to relevant and irrelevant parts of the stimulus. We have proposed a network based theory of these same phenomena [Sandon 1989,1990], which combines aspects of both feature integration and stimulus similarity, while providing a framework for achieving computational efficiency.

In this paper, we discuss a particular aspect of our theory, which is most relevant to Duncan's similarity based theory. In particular, while Treisman's theory involves discrete distinctions between parallel and serial visual search, Duncan's theory involves a continuously varying dependence of search times on display size. In our network model of attention, this dependence is explained in terms of feature maps with mutually inhibiting activations and the time course of the winner-take-all (WTA) mechanism responsible for selecting the attentional focus.

In the next section we review the details of the feature integration and stimulus similarity theories, and mention related network models of attention. The following sections describe our own network model and the results of some relevant computer simulations.

Background

Feature integration theory. Treisman and her colleagues [Treisman & Gelade 1980; Treisman & Schmidt 1982] have collected data on human visual performance for a variety of tasks where attention is implicated. In particular, her data suggest that in a visual search task, when the target to be detected differs from non-target distractors along a single primitive feature dimension, the target can be detected in an amount of time that does not depend on the number of distractors. Thus, it appears that the search for the target proceeds in parallel, with all objects, target and non-target alike being processed simultaneously. This is referred to as feature search. On the other hand, when the target is distinguished from the distractors by a combination of values in two feature dimensions, the amount of time required to detect the target increases linearly with the number of distractors in the input. Thus, it appears that the search for the target proceeds serially, with all objects being processed in sequence. This is referred to as conjunction search.

Stimulus similarity theory. Duncan provides an alternative theory to explain the reaction time dependencies of visual search [Duncan 1985; Duncan & Humphreys 1989]. Setting aside the parallel versus serial distinction, Duncan claims that the search times depend only on two measures: the similarity between the target and the nontarget distractors (T-N similarity), and the similarity among the nontargets (N-N similarity). Higher T-N similarity increases the search time for the target, while lower N-N similarity has an even more pronounced effect in increasing the search time. This is referred to as the stimulus similarity theory. Duncan also emphasizes the importance of object size in obtaining the reported attentional effects. In particular, the search time dependence on number of objects is itself dependent on the ratio of object size to retinal eccentricity of the object within the image.

Previous connectionist models. Prior to this work, a handful of network models have been proposed to explain individual pieces of the attentional data. Hinton and Lang [1985] used a network similar to Hinton's [1981] mapping network to simulate illusory conjunctions. Using a winner-take-all (WTA) array of processing elements to represent the attentional focus, they found that illusory conjunctions could be made to occur if a random input pattern was presented prior to settling of the WTA process in the attention array. Mozer [1988] used a similar network structure to simulate a probabilistic attentional mechanism having variable focus size in his MORSEL system. Sandon & Uhr [1988] used a network similar to Hinton's mapping network, but implemented the representation of location hierarchically, as a means of more efficiently representing and learning translation invariant object recognition. This hierarchical representation of location leads naturally to the idea of representing the attentional focus in a hierarchy.

The network model

We have designed a network model of visual attention based on constraints drawn from three fields: computational principles developed in the machine vision literature, knowledge of the neurophysiology of vision, and behavioral data. In this section, we summarize the overall model. Sandon [1990] discusses the considerations motivating the network design, and presents the results of other simulation experiments.

The underlying structure of the network is based on the original network used by Hinton [1981] to model translation (as well as rotation) invariant object recognition. The idea is to represent both shape features and spatial location of

objects in separate arrays, and to combine these two sources of information multiplicatively (using so-called conjunctive connections) in recognizing objects. This design provides a computational structure that allows for the representation of multiple competing hypotheses about the location and identity of objects, and produces an interpretation of the image which is most consistent with the dual set of constraints (location features and shape features).

In previous work [Sandon & Uhr 1988], we augmented a pyramid structured shape network with a location subnetwork to efficiently perform translation-invariant object recognition. We used a two layered hierarchical representation for the location subnetwork, which reduced the required connectivity, allowing us to train the network to recognize objects in given positions and then generalize to novel positions. The model proposed here uses a hierarchical representation of spatial location as the basis for an attentional mechanism. This structure automatically provides translation-invariant recognition of objects, since its underlying structure is that of the translation-invariant network. To this basic structure, we add a capability for multiple scale analysis, by providing separate pathways for processing the image at different levels of resolution.

For the purpose of recognizing relatively simple geometric shapes, as are generally used as stimuli for psychological and neurophysiological studies, a shape feature hierarchy such as that used by Uhr [1978] or Sabbah [1985] is appropriate. Since the evidence for various shape feature detectors in the human visual system is still a matter of debate [Treisman & Gelade 1980; Sagi 1988; McLeod, et. al. 1988; Duncan & Humphreys 1989], this model makes no *a priori* commitment to a particular set of features. Instead, we develop our feature set incrementally as we simulate more of the behavioral data.

We must also specify the set of features used to activate the attention layers. There are two aspects to be considered in addressing this problem. First, what are the attentional features themselves, and second, how do these features interact to produce the attentional activation? Regarding the features themselves, we again choose those that produce simulation results in agreement with the behavioral data. At the lowest layer of attention, oriented edges and lines are used. At the higher layer, perceptual grouping features, such as parallel and collinear lines, symmetry, and adjacent line terminations are appropriate [Lowe 1987; Witkin & Tenenbaum 1983]. To combine features for attentional input, we implement an interaction among like features prior to their introduction to the attention array. In particular, a central-excitatory, peripheral-inhibitory interaction is applied to each of the feature maps used as input to the attention array. This contrast enhancement of features produces input to the attention array only when a given feature occurs in the image in relative isolation from other features of the same type.

The attentional focus is determined by a WTA competition within the attention array. The effect of the attentional activity is to gate the features from a particular region of the image up to higher layers of the network, where object recognition occurs. As previously noted, the features comprising the input to this recognition process are location invariant. Similarly, these features are made scale invariant, by transforming each possible scale to a normalized size. The individual data paths representing the different processing scales are combined prior to recognition processing using a policy of global precedence [Hughes et. al. 1990]. This policy assures that, in the absence of other information, the lowest resolution data path that exhibits significant attentional activity is gated to the higher processing levels.

Figure 1a summarizes the attentional model described in this section. The leftmost data path is for fine scale processing and includes two levels of attention. Data paths are bidirectional, providing a pathway for attentional priming, and other task-directed responses. The middle data path is similar, but starts with a lower resolution intensity image, and requires only one attention layer to select features for processing by the recognition processor. The rightmost data path involves the coarsest resolution intensity image, whose features are passed directly to the recognition processor. These three data paths provide processing at three scales. The choice of which scale to process is made by the scale arbitrator, which implements the global precedence policy.

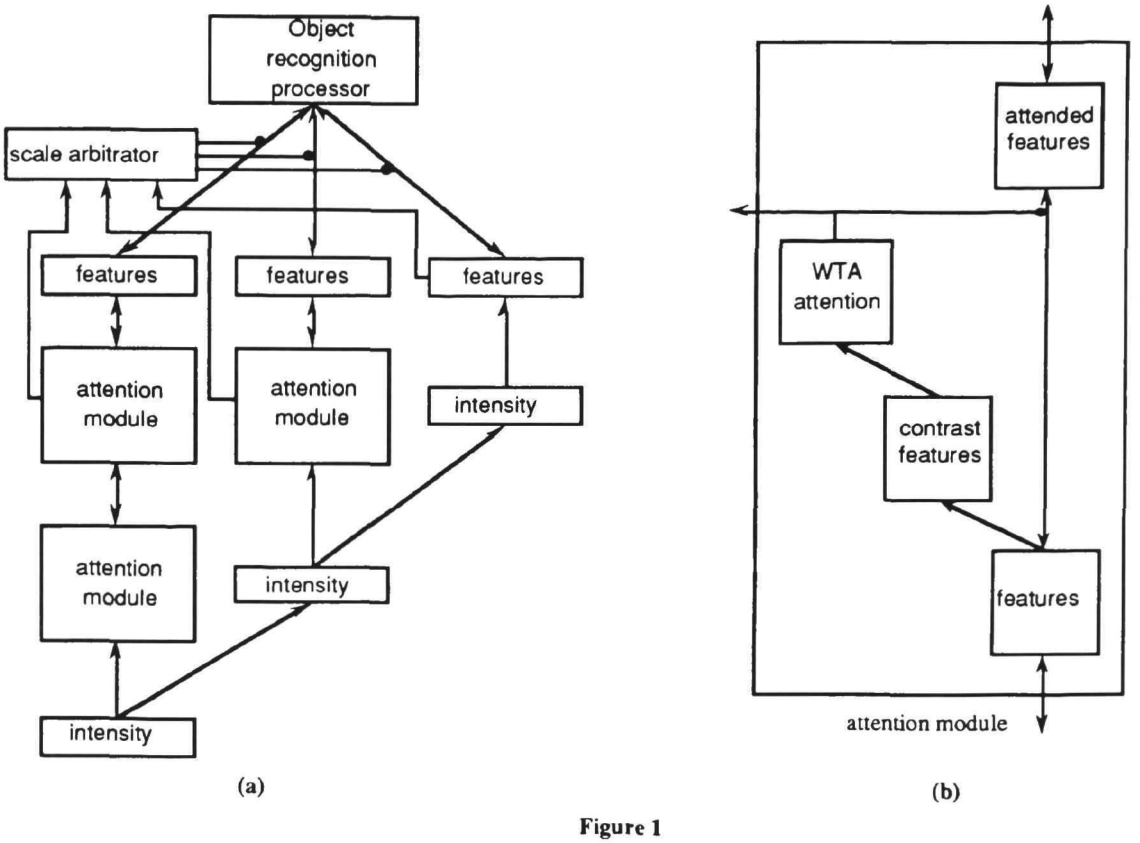


Figure 1

Simulation results

We have simulated a number of the experiments described by Duncan and Humphreys [1989] using a subnetwork of the model just described. The simulated network uses a 64 x 64 pixel intensity array to represent the input stimulus. We use only a single layer attention mechanism, but at two different scales. The primitive feature maps implemented in the simulation are four orientations of lines, and four orientations of 'L' corners. Each primitive feature map has an associated contrast feature map, which is computed by applying an on-center, off-surround lateral inhibition operator to the feature map. The activations of all contrast features are summed to provide the bottom-up initial activation of the attention arrays. To simulate the task-directed component of attentional activation, we weight the contributions of feature arrays associated with the known target object more than the remaining feature arrays when summing them. A WTA operation is applied to the

initial attentional activation, based on the following inhibition rule [Koch & Ullman 1985]:

$$dy_i / dt = y_i (x_i - \Sigma x_j y_j)$$
$$y_i (0) = (1/N) + \eta \quad ; \quad \eta \text{ is zero mean noise}$$

In this rule, the x_i are the attention layer activations, while the y_i are the activations of a set of auxiliary nodes. The rule constrains the y_i always to sum to 1. We have used a termination criterion that requires one of the y_i to exceed a value of 0.6, which is sufficient to guarantee that the corresponding region will win the competition. We report the number of WTA iterations required to reach criterion, as an indicator of the response time of the network. We do not include the recognition subnetwork in the simulation.

Duncan describes four extreme cases with respect to his similarity measures:

Case	T-N similarity	N-N similarity	dependence on display size
A	low	high	low
B	high	high	intermediate
C	low	low	low
D	high	low	high

We have simulated the experiments reported in [Duncan & Humphreys 1989] for each of the above four cases, using display sizes of 4 and 6. In addition, we have repeated these experiments for display sizes of 15 and 20. The displays in Figure 2 show the stimulus pattern(i) and initial attention activation(ii) for the following experiments: (a) Case A, Size 15; (b) Case B, Size 6; (c) Case C, Size 4; (d) Case D, Size 20. In the table below, we report the number of iterations of the WTA rule applied to the attention layer, required for one of the auxiliary nodes to exceed the value 0.6. The first value corresponds to the high resolution attention array, the second value to the low resolution array. We take the response time to be the lower of the two values.

	Display Size			
Case	4	6	15	20
A	22 - 15	27 - 15	12 - 15	12 - 13
B	22 - 17	27 - 18	24 - 99+	25 - 89
C	22 - 13	27 - 13	13 - 15	13 - 13
D	22 -*17	27 -*19	36 - 99+	*15- 99+

Discussion

The simulation results for this subnetwork are in agreement with the experimental data reported by Duncan and Humphreys. For Case A, our model achieves a search time independent of display size due to the strong response in the 'L' corner feature map corresponding to the target, which is directly reflected in the attentional activation, and the inhibition among the nontargets in the other feature maps, due to their high similarity. Case C gives similar results, though the inhibitory interactions among nontargets is weaker, due to their distribution among multiple feature maps. These two cases are not distinguished in Treisman's theory, both corresponding to the feature search condition.

For Case B, we get a moderate dependence of search time on display size. This is due to the somewhat weaker activation of the attention array corresponding to the

target that results from the inhibitory interaction of the target and nontargets in the feature maps. Finally, for Case D, the difference in attentional activity corresponding to target and nontargets is low, as a result of weaker inhibitory interactions among nontargets due to their low similarity, and of strong interactions between target and nontargets due to high similarity. This lack of contrast in some cases leads the WTA procedure to choose a nontarget as the selected focus of attention (as indicated by an asterisk in the table). This behavior would, in a more complete simulation, lead to a truly serial search for the target, mediated by multiple WTA settlings with inhibitory tagging [Klein 1988] between settlings. Again, Treisman's theory does not distinguish these cases, but presumes a mechanism like the multiple settlings as the source of the serial search times.

The connectionist network described above is intended to provide an alternative model of attentional behavior to that characterized by Treisman or by Duncan. We now consider the relations among the three theories. Treisman's theory appears to be a special case of Duncan's, since her feature-conjunction search dichotomy is subsumed by Duncan's T-N similarity measure, while the N-N similarity measure allows additional data to be modelled. On the other hand, despite Duncan's insistence that visual search times depend not on which features in target and nontarget objects are primitive, but only on the similarity among these objects, our model suggests that this 'similarity' measure is highly dependent on the identity of the primitive features. Since the interactions among features occur within the primitive feature maps, it is in the primitive feature dimensions that similarity is determined.

Treisman identifies two different search complexities, parallel and serial, which correspond to search times that are independent of display size and linearly dependent on display size, respectively. Duncan rejects these distinctions, preferring to describe a varying dependence of search time on display size. In our model, we observe three different search modes. When the target produces a strong activation in the attention layer, while distractors produce weak or no activation, the WTA procedure converges to an isolated activation corresponding to the target in a time that is virtually independent of the number of distractors. This corresponds to parallel search. When the target and nontargets produce approximately equal attentional activations, the WTA procedure converges much more slowly. More importantly, the region selected is as likely to correspond to any image object as any other, so multiple WTA settlings may be required to locate the target. This corresponds to serial search. When the target produces an attentional activation that is only moderately stronger than that produced by distractors, the dependence of search time on display size is determined by the WTA procedure. The resulting constraint propagation, or relaxation, search is parallel, in that it considers all object locations simultaneously, but is dependent on both the number and magnitude of the non-maximal activations, yielding a display size dependency that is different than either of the other two cases.

Finally, we mention some additional experiments reported by Duncan & Humphreys. In addition to the similarity measures, another critical determinant of search times is the size of the objects in the display. For a given eccentricity, larger objects reduce the display size dependency. In our model, this result is predicted by the reduced interaction of objects within the feature maps, due to the limited extent of the inhibitory connections. In another experiment which used 'L' corners of different orientations for targets and nontargets, a large dependence of search time on display size was found when the nontargets were CW and CCW 90° rotations of the target. The explanation given by Duncan is that the target is similar to the

nontargets, being the same object except for a rotation. Our current simulations could not produce this result, since there is no representation of rotation, nor of similarity across rotation. Our model would have to be elaborated to include rotation, as in Hinton's [1981] original model, in order to simulate this behavior.

References

Duncan, J., "Visual search and visual attention," in *Attention and Performance XII*, ed. M. I. Posner & O. S. M. Marin, pp. 85-105, Erlbaum, Hillsdale, NJ, 1985.

Duncan, J. and G. W. Humphreys, "Visual search and stimulus similarity," *Psychological Review*, vol. 96, pp. 433-458, 1989.

Hinton, G. E., "Shape representation in parallel systems," *Proc. 7th IJCAI*, pp.1088-1096, Vancouver, 1981.

Hinton, G. E. and K. J. Lang, "Shape recognition and illusory conjunctions," *Proc. 9th IJCAI*, vol. 1, pp. 252-259, Los Angeles, August 1985.

Hughes, H. C., R. Fendrich and P. A. Reuter-Lorenz, "Global versus local processing in the absence of low spatial frequencies," to appear, *J. Cognitive Neuroscience*, 1990.

Klein, R., "Inhibitory tagging system facilitates visual search," *Nature*, vol. 334, pp. 430-431, August 1988.

Koch, C. and S. Ullman, "Shifts in selective visual attention: toward the underlying neural circuitry," *Human Neurobiol.*, vol. 4, pp. 219-227, 1985.

Lowe, D. G., "Three-dimensional object recognition from single two-dimensional images," *Artificial Intelligence*, vol. 31, pp. 355-395, 1987.

McLeod, P., J. Driver and J. Crisp, "Visual search for a conjunction of movement and form is parallel," *Nature*, vol. 332, pp. 154-155, March 1988.

Mozer, M. C., "A connectionist model of selective attention in visual perception," *Proc. 10th Conf. Cognitive Science Society*, pp. 195-201, Montreal, August 1988.

Sabbah, D., "Computing with connections in visual recognition of origami objects," *Cognitive Science*, vol. 9, pp. 25-50, 1985.

Sagi, D., "The combination of spatial frequency and orientation is effortlessly perceived," *Perception & Psychophysics*, vol. 43, pp. 601-603, 1988.

Sandon, P. A., "Simulating visual attention," to appear, *J. Cog. Neuroscience*, 1990.

Sandon, P. A., "An attentional hierarchy," commentary on "A solution to the tag-assignment problem of neural networks" by G.W. Strong and B. A. Whitehead, *Behavioral and Brain Sciences*, p. 414, September 1989.

Sandon, P. A. and L. M. Uhr, "An adaptive model for viewpoint-invariant object recognition," *Proc. 10th Conf. Cognitive Science Society*, pp. 209-215, Montreal, August 1988.

Treisman, A. and H. Schmidt, "Illusory conjunctions in the perception of objects," *Cognitive Psychology*, vol. 14, p. 107-141, 1982.

Treisman, A. and G. Gelade, "A feature-integration theory of attention," *Cognitive Science*, vol. 12, pp. 99-136, 1980.

Uhr, L. M., "Recognition cones and some test results," in *Computer Vision Systems*, ed. A. R. Hanson and E. M. Riseman, pp. 363-377, Academic Press, New York, 1978.

Witkin, A. P. and J. M. Tenenbaum, "What is perceptual organization for?," *Proc. IJCAI-83*, pp. 1023-1026, Karlsruhe, West Germany, August 1983.

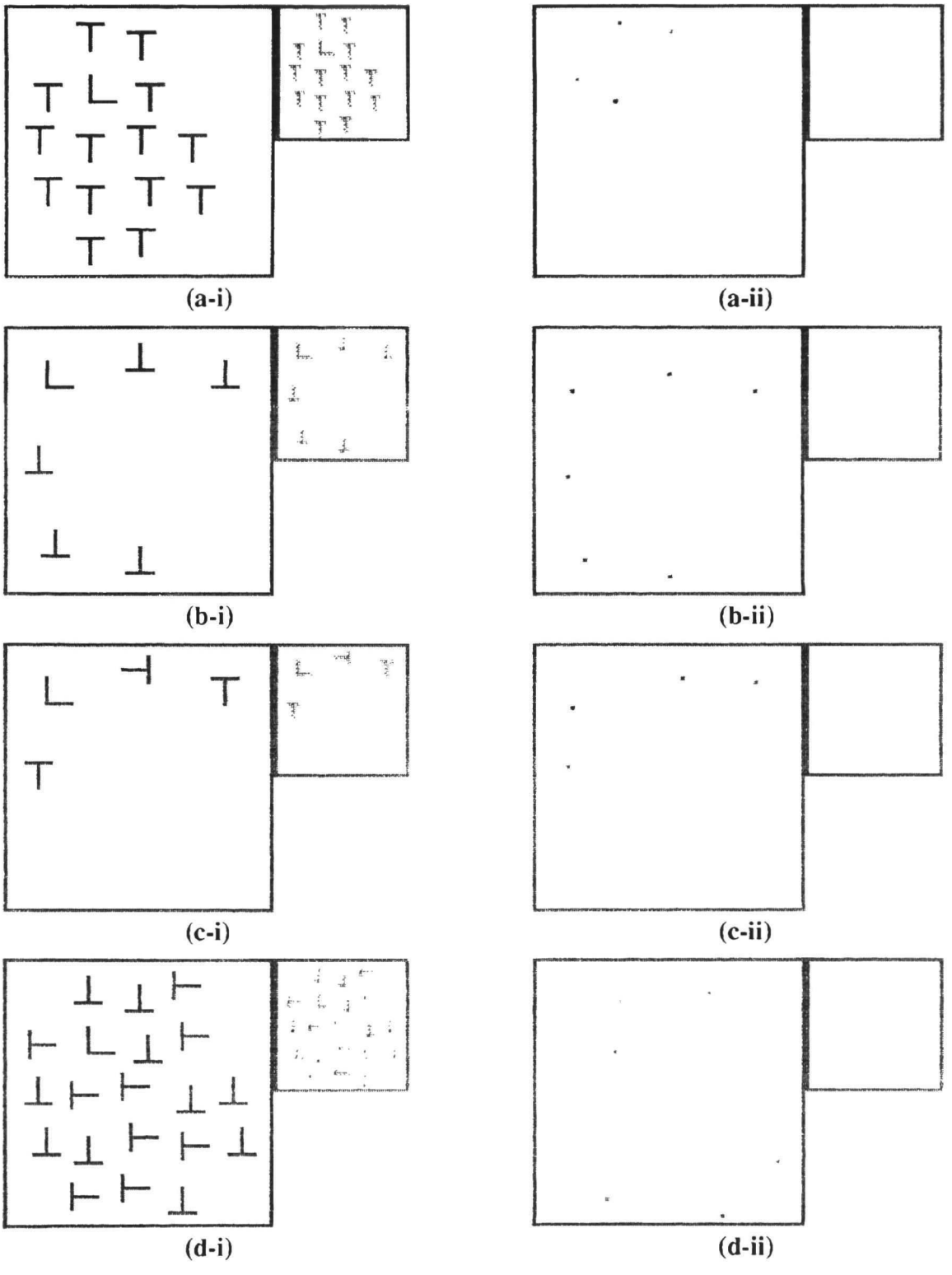


Figure 2

A Computer Model of 2D Visual Attention*

Mark Wiesmeyer and John Laird
Artificial Intelligence Laboratory
The University of Michigan

In this paper we present a model of human visual attention that is an extension of a preexisting cognitive theory, the Model Human Processor (MHP) [Card *et al.*, 1983]. The type of visual attention that we model is independent of eye movements and serves as a form of stimulus selection within the visual field. Our goal is to provide a finer grain of reaction time prediction for visual tasks requiring attention than is provided by the MHP. We have developed algorithms based on our model to account for response times of object identification experiments of Colegate, Hoffman and Eriksen (1973) (hereafter CHE). We have implemented these algorithms in Soar [Laird *et al.*, 1987; Laird *et al.*, 1990], which is a reification of some of the basic principles of the MHP, and has been proposed as a unified theory of cognition [Newell, 1990]. Our system models the sequencing of deliberate controllable visual acts of cognition that take on the order of 50 msec. The results of our work suggest that a variant of the "Zoom Lens Model" [Eriksen and Yeh, 1985; Eriksen and St. James, 1986] of visual attention coupled with Soar's theory of deliberate behavior is sufficient for modeling these phenomena, thus, extending the predictive power of the Model Human Processor.

1 The Model Human Processor

The MHP is a cognitive model of human behavior that has been most successfully applied to modeling the timing of man and machine interactions [John, 1988; John and Newell, 1989]. The goal of the MHP is to serve as an engineering tool for estimating human performance. Figure 1 shows a simplified representation of the MHP from Card et al. (1983). On the top left-hand side of the figure is the Perceptual Processor or Perception, which consists of sensory modalities, such as vision and audition. (For economy of space, the figure only shows vision.) Each modality delivers a limited amount of information at a particular rate to Working Memory. On the bottom left-hand side of the figure is the Motor Processor or Motor, while on the right-hand side of the figure is Cognition, which consists of Working Memory, Long-Term Memory, and the Cognitive Processor. In the MHP, tasks are decomposable into deliberate actions that take place in the form of discrete sequential operator applications in Cognition. Some operators may require input from the Perception, others may initiate external acts using Motor, and finally, some may use existing data in Working Memory for internal calculations that are independent of Perception and Motor.

Each subsystem in the MHP has a range of nominal cycle times. The cycle times that we use in our model for Perception and Motor are MHP median figures. The cycle time we use for Cognition is taken from John (1988) and is 20 msec faster than the MHP median figure for Cognition, but well within the range of nominal cycle times for Cognition in the MHP.

Perceptual (vision) cycle time (τ_p)	= 100 msec	(Range: 50-200 msec)
Cognitive cycle time (τ_c)	= 50 msec	(Range: 25-170 msec)
Motor cycle time (τ_m)	= 70 msec	(Range: 30-100 msec)

The total time required to perform a task is calculated by first creating an algorithm for performing the task using operators. Using the cycle times of each processor to compute the duration of each operator, the total time to complete the algorithm can be calculated. The three processors can run in parallel, so, for instance, once a motor command has been initiated, other cognitive operators can be applied at the same time, as long as they are not dependent on the result of the motor operator.

*This research was sponsored by grant NCC2-517 from NASA Ames.

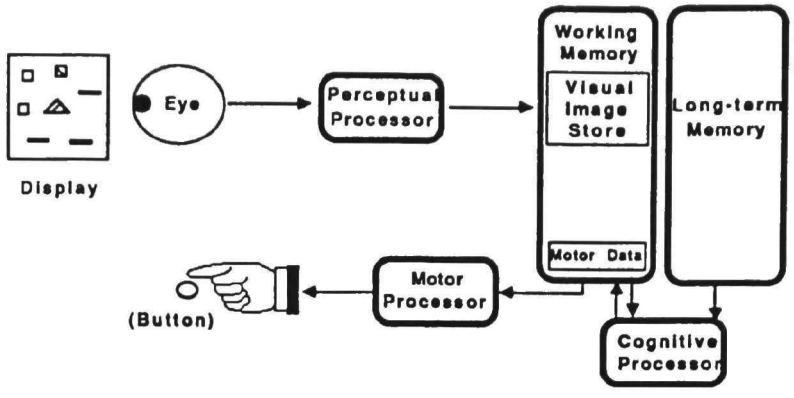


Figure 1: The Model Human Processor

We are using Soar [Laird *et al.*, 1987] as the underlying architecture for implementing our operator sequences. Soar has all of the basic subsystems of the MHP, including Perception, Cognition, and Motor.[1] In Soar there is a Working Memory and a Long-term Memory, and deliberately selected operators provide the basis of action. Soar's Long-term Memory is a parallel production system. Perception is implemented in Soar as Lisp functions that transduce environmental stimuli and send input to working memory, while Motor is implemented in Soar as Lisp functions that receive output from Working Memory and then act on the environment. In Soar, low-level Perception occurs in parallel with the firing of productions, as does Motor, once it has been initiated through an operator application.

The basic deliberative act in Soar is the selection of an operator to apply. Thus the operators in the MHP map directly onto operators in Soar. Both the selection and application of operators are performed by productions matching against Working Memory and suggesting changes to it. Thus, productions control which operators are selected, and once selected, how they apply. In Soar, specific operators (that is, instantiations of operator types) are created as soon as the data needed to instantiate them are available. In general, this means that new operators will be created, and ready for selection, during the application of the currently selected operator. However, only one operator can be applied on any given cycle. Although Soar is programmed at the level of productions, operators are the appropriate level for describing algorithms that correspond to processing in the MHP.

2 A Model of Visual Attention

The goal of our work is to develop a complete, computational model of visual attention. Unfortunately, there is no existing computational theory which covers all visual attention phenomena. There are many experimental paradigms with accompanying data that provide constraints for a complete model. In this paper we will model three general phenomena: humans are able to control the portion of the visual field that they attend to [Sperling, 1960]; attention is required for object identification [Treisman and Gelade, 1980; Treisman and Schmidt, 1982]; and precuing facilitates object identification [Colegate *et al.*, 1973]. These are critical visual attention phenomena which are all observed in letter-wheel task of CHE. We will use this task to demonstrate the details of the model. Our implementation in Soar has also been used to simulate other phenomena that have been attributed to visual attention, including illusory conjunctions and various reaction time behaviors associated with search [Wiesmeyer and Laird, 1990], that are beyond the scope of this paper.

Figure 2 shows how we have extended the MHP to include visual attention. On the top left-hand side of the figure again is Perception, but it has now been expanded to show details of how it delivers visual data to Working Memory. A single ovoid *region of attention* controlled by Cognition, as in Eriksen's "Zoom Lens

[1] Soar has additional capabilities for planning and learning that are not needed for the tasks described in this paper.

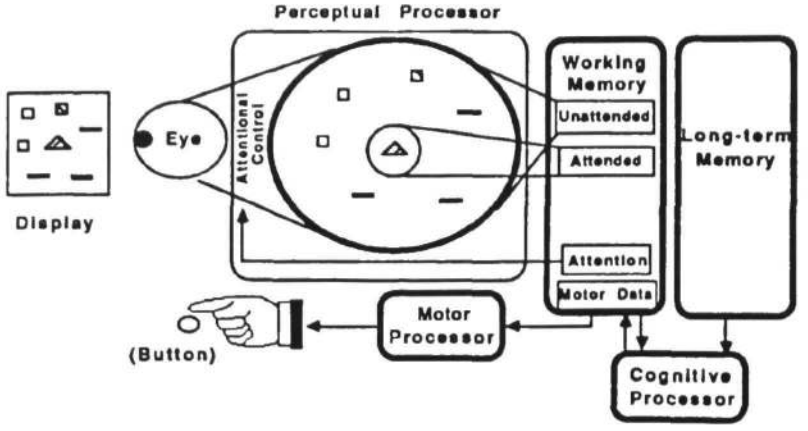


Figure 2: The model human processor with visual attention extension

Theory" [Eriksen and Yeh, 1985; Eriksen and St. James, 1986], separates the visual field in Perception into two regions: attended and unattended. The region of attention is under control of Cognition and can be moved around the visual field. Perception, as in the MHP, converts stimuli from these regions into symbols that are delivered to Working Memory; we call these symbols *features* after Treisman [Treisman and Gelade, 1980; Treisman, 1987]. Thus, input to the Visual Image Store in Working Memory is separated into two sets: *attended features* from the attended region and *unattended features* from the unattended region.

In our model, features are of two types: shape and color, although color is not needed to model the letter wheel task. Motion (change) information is associated with features, when appropriate, and features can be *marked* by productions. Marking allows for a simple memory of whether a feature has been attended. Except for motion information and marks, features have no other explicit properties besides their identities.

Attention is controlled through the `Feature-Shift` operator. A separate operator is created for each unmarked feature in the visual field and application shifts the region of attention to that feature. In shifting attention, the new region of attention may also include other features that are near the selected feature. The `Feature-Shift` operator can be further specialized to shift to the *nearest* feature of a given type. For example, `Feature-Shift` can be used to shift to the nearest new shape feature.

An important property of our model is that attention improves the resolution of shape features. We have adopted a simplified version of Eriksen's "Zoom Lens Model" of attention, which theorizes that unattended stimuli are always lower in resolution than attended stimuli and that there is an inverse relationship of region extent and attended feature resolution. Low resolution means, for instance, that a letter stimulus that does not appear in the region of attention or appears in a relatively large region of attention, might appear as a unidentifiable "blob" in Working Memory.

3 The Letter Wheel Experiment

We are using Colegate, Hoffman and Eriksen (1973) as a prototypical example of an object identification task that involves precuing. Other experiments have similar results under slightly different conditions [Eriksen and Hoffman, 1972; Eriksen and Hoffman, 1973]. In CHE, subjects were presented with displays similar to those shown in Figure 3. The task was to identify the cued letter. Subjects responded vocally. Letters from the set `A,E,M,U` were systematically distributed over the various locations in 8 and 12 letter displays so that the effects of letters neighboring the target on reaction time could be studied. Displays were 2 degrees in visual angle and presented with a tachistoscope—2 degrees is the extent of the fovea and approximately the diameter of the entire letter wheel on the right of the figure if this paper is held at arm's length.

There were two basic modes of stimulus delivery, simultaneous presentation and precued presentation. In both modes subjects were told to fixate on a fixation point until a cue for the location of a letter to be identified was presented. In simultaneous presentation the cue and the letter arrived at the same time, while

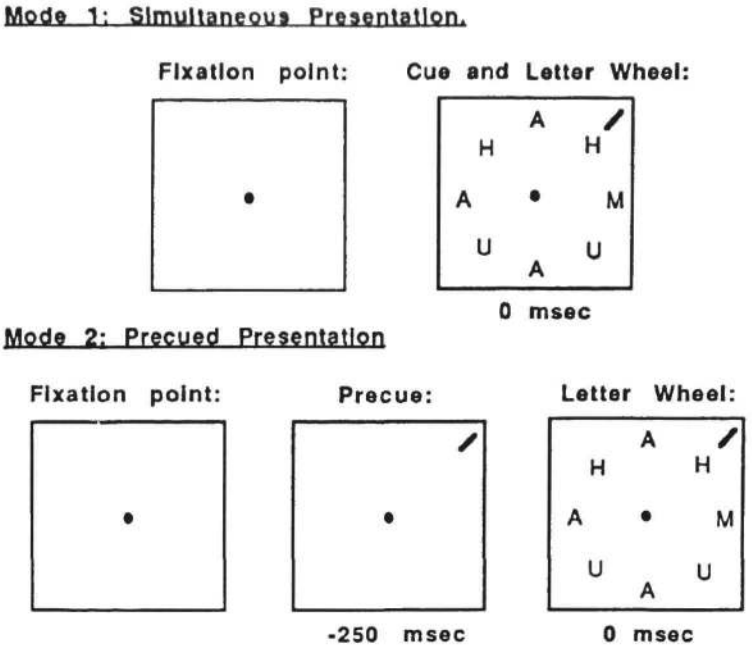


Figure 3: Example letter wheel stimuli

in precued presentation the cue preceded the letter wheel by an interval of up to 350 msec. In both modes, subjects were required to identify the cued letter. CHE's data showed that reaction times improve at a constant rate as the precue period increases up to 250 msec. With a 250 msec precue there was a 100 msec improvement in reaction time with respect to the simultaneous presentation condition. Response times for 8-letter displays were always slightly faster than for 12-letter displays. We do not account for the differences between the 8 and 12 letter displays in this paper, but have concentrated on getting the response times of our simulations within the range of observed data.

4 Modeling the Letter Wheel Experiment

To model these experiments requires three operators in addition to `Feature-Shift`:

`Identify` : Identifies an attended object based on the shape feature (cue and target in our simulations).
`Verify` : Verifies that an object identified (the cue in our simulations) was the one expected.
`Respond` : Performs the appropriate motor action.

These operators form the minimal set we could find to perform the task given the level of computation assumed to be possible in the supporting productions. The only questionable one might be `Verify`. This is used to ensure that the result of a `Feature-Shift` and subsequent `Identify` was correct. Without it, the system would be more prone to errors.

We assume that τ_c is the time required for an average cognitive operator to be created or selected and applied—creation being separate from selection and application; τ_p is the time required for a stimulus to get from the retina to working memory; and τ_m is the time required for a new Working Memory element to effect a motor command. Thus `Identify` and `Verify` take 100 msec for creation, selection, and application, while `Respond` takes an additional 70 msec (170 msec total) because of the motor cycle time.

The application of `Feature-Shift` involves interaction between Cognition and Perception, which makes determining its time course problematic. We decompose its time course into the three stages required for Cognition to (1) react to a new stimulus, (2) shift visual attention to that new stimulus, and (3) then receive new features in Working Memory that reflect the change in visual attention. The only unknown component of this series of events is the time required for a visual stimulus to travel from the locus of action of attention

to Working Memory; it can be estimated from brain data: Visual attention has been shown to act at a location in the brain called V4 [Moran and Desimone, 1985]. From V4 attended stimuli go to the inferior temporal cortex and then to the amygdala and the hippocampus [Mishkin and Appenzeller, 1987]. The amygdala and hippocampus are known to be involved in visual memory and are thus assumed to be the beginnings of Cognition (Working Memory). Since V4 is physically located about half way between the retina and the amygdala and hippocampus, we estimate that it should take about 1/2 of τ_p (or 50 msec) for the effects of visual attention to be reflected in the contents of Working Memory. Thus, we estimate the amount of time required for the entire process as:

100 msec	τ_p for propagation of data from the retina to working memory
50 msec	τ_c for `Feature-Shift` creation
50 msec	τ_c for `Feature-Shift` selection and application
50 msec	$\frac{1}{2}\tau_p$ for propagation of attended data from V4 to working memory
250 msec total	

For features already in Working Memory, the time required to shift attention to them and then receive new data reflecting the change in attention is the sum of the last three steps above or 150 msec. Admittedly, it would be preferable to use an empirically derived time in our model rather than these estimates, but we have not been able to locate a suitable one in the literature and have found this time to work well in our algorithms.

Bearing these figures in mind and the fact that next operator creation can overlap with current operator application, the job at hand in creating an algorithmic cognitive model is to fit operators together so that the requisite task may be done and known time constraints may be satisfied.

We have developed algorithms for both the simultaneous presentation and 250 msec precue conditions. Observed mean response times from CHE are 574 msec (483 - 635 msec) for the simultaneous presentation condition and 491 msec (427 - 569 msec) for the 250 msec precued presentation condition. Our algorithms predict 570 msec and 470 msec, respectively, and are thus both within the range of times obtained from experimental subjects. Additionally, the improvements due to precuing predicted by our model (100 msec) correlate closely with those of actual subjects (mean 83 msec)[2]

Details of the both algorithms follow. Both are implemented in Soar using a single set of productions that controls the selection of operators based on the contents of Working Memory—there is no explicit selection strategy programmed for the two tasks. In these traces, the selection of operators is implied and occurs for each operator immediately preceding its application.

In the simultaneous presentation trace in Figure 4, a single `Feature-Shift` is performed to the cue. Both the cue and the letter are in the region of attention and their features are in the new attentional region at 200 msec. The algorithm next identifies and verifies that the cue is in the attentional region (250 - 400 msec) and then identifies the letter (400 - 450 msec) and then responds (450 - 570 msec).

In the 250 msec precue case shown in Figure 5, a `Feature-Shift` is applied to shift to the cue. From start to finish, this takes 250 msec (-250 - 0 msec). Following the shift, the cue is identified and verified (0 - 100 msec). At the same time, the letter wheel is being processed by perception and becomes available in Working Memory at 100 msec. At this point, a second shift is performed to the nearest letter (150 - 250 msec) followed by its identification (250 - 300 msec) and finally the response (350 - 470 msec).

In the simultaneous presentation condition, only one shift of attention is required, while in the precued condition two shifts are required. In spite of this, the precued condition is still faster. There are two reasons for this: (1) The first shift of attention for the precued condition finishes before the wheel is presented and timing starts. (2) The `Identify` and `Verify` operators for the cue occur during the period in which the wheel stimulus is traveling from the retina to working memory (labeled "overlap" in the second algorithm).

The CHE results show that there is no improvement in reaction time when precuing exceeds 250 msec. The response time of our implementation does not improve either, since the shift to the target letter is contingent upon its shape feature appearing in Working Memory, which time will never change from 100 msec after the stimulus is presented.

[2]There is a great deal of variation in experimental results for precuing in object identification experiments. The CHE experiment was chosen simply on the basis of its typicality.

Time	Operator Event	Event/Comment
		Attention centered on fixation point
0 msec	None	Wheel and cue at retina
50 msec	None	Wheel and cue in P
100 msec	Feature-Shift created	Wheel and cue in WM
150 msec	Feature-Shift applies	Shift to cue and letter
200 msec	Feature-Shift completed	Attentional message at V4
250 msec	Identify (cue) created Identify (letter) created	Cue and letter attended in WM
300 msec	Identify (cue) applies Verify (cue) created	
350 msec	Verify (cue) applies	
400 msec	Identify (letter) applies Respond created	Letter identified
450 msec	Respond applies	
500 msec	Respond completed	Motor output begins
570 msec		Motor command completed

Figure 4: Operator trace of the simultaneous presentation of the letter wheel and cue

The results of the model for precuing between 0 and 250 msec are less clear. The CHE results show there is incremental improvement as the precue goes from 0 to 250 msec (in 50 msec increments). Between 0 and 150 msec the predictions of our model are unclear because the letter wheel is being processed by Perception while attention is being shifted. If we assume that the time it takes Perception to process data is stochastic, then the increase in precue would increase the probability that the precue would be available soon enough to allow the extra shift.

5 Discussion

In this paper, we have developed a computational model of visual attention and algorithms that are sufficient for calculating the reaction times of a simple task that requires visual attention. In the process, we have estimated the time for Cognition to react to a new stimulus, shift visual attention to that new stimulus, and receive new features in Working Memory that reflect the change in visual attention.

The exact results of this model are dependent on many assumptions including the cycle times for Perception, Cognition, Motor, and Attention. However, these assumptions are not unique to this model; all except the time for changes in attention were based on previous research. The set of operators that we have chosen is another key assumption. A challenge for our future modeling is to see if these operators are sufficient for additional tasks requiring visual attention.

Time	Operator Event	Event/Comment
		Attention centered on fixation point
-250 msec	None	Cue at retina
-200 msec	None	Cue in P
-150 msec	Feature-Shift created	Cue in WM
-100 msec	Feature-Shift applies	Shift to cue
-50 msec	Feature-Shift completed	Attentional message at V4
0 msec	Identify (cue) created	Cue attended in WM / Wheel at retina
50 msec	Identify (cue) applies Verify (cue) created	Wheel in P
100 msec	Verify (cue) applies Feature-Shift(s) created	Wheel in WM
150 msec	Feature-Shift applies	Shift to nearest letter
200 msec	Feature-Shift completed	Attentional message at V4
250 msec	Identify (letter) created	Letter attended in WM
300 msec	Identify (letter) applies Respond created	Letter identified
350 msec	Respond applies	
400 msec	Respond completed	Motor output begins
470 msec		Motor command completed

Figure 5: Operator trace of the 250 msec precue

A major simplification in the current implementation is that it is unnecessary to model the inverse relationship between region size and shape stimulus resolution strictly. Instead, we assumed that unattended shape features were simply not identifiable, but they were detectable. This allowed them to act as destinations in the visual field for shifts of attention. Once a feature was attended, its identity was available through Working Memory; that is, an `Identify` operator could be created for it. We may need to refine our interpretation of the "Zoom Lens Model" so that identifiability takes on a more stochastic quality based on proximity to the region of attention instead of the discrete quality it now has. It is possible that this stochastic quality could help to account for noise effects, such as those found in CHE, where distracting letters near the target slowed identification response time.

Although extensions are necessary, the results of this work show that the augmented MHP can be used to account for a limited group of visual attentional phenomena, and that Soar is a sufficient symbolic architecture for building running MHP models.

References

[Card *et al.*, 1983] S.K. Card, T.P. Moran, and A. Newell. *The Psychology of Human-Computer Interaction.* Erlbaum, Hillsdale, NJ, 1983.

[Colegate *et al.*, 1973] R. Colegate, J.E. Hoffman, and C.W. Eriksen. Selective encoding from multielement visual displays. *Perception and Psychophysics*, 14:217–224, 1973.

[Eriksen and Hoffman, 1972] C.W. Eriksen and J.E. Hoffman. Some characteristics of selective attention in visual perception determined by vocal reaction time. *Perception and Psychophysics*, 11:169–171, 1972.

[Eriksen and Hoffman, 1973] C.W. Eriksen and J.E. Hoffman. The extent of processing noise elements during selective coding from visual displays. *Perception and Psychophysics*, 14:155–160, 1973.

[Eriksen and St. James, 1986] C.W. Eriksen and J.D. St. James. Visual attention within and around the field of focal attention: A zoom lens model. *Perception and Psychophysics*, 40:225–240, 1986.

[Eriksen and Yeh, 1985] C.W. Eriksen and Y.Y. Yeh. Allocation of attention in the visual field. *Journal of Experimental Psychology*, 11:583–597, 1985.

[John and Newell, 1989] B.E. John and A. Newell. Cumulating the science of HCI: From S-R compatibility to transcription typing. In *Proceedings of CHI*, pages 109–114, Austin, Texas, April 30-May 4 1989.

[John, 1988] B.E. John. *Contributions to engineering models of human-computer interaction.* PhD thesis, Carnegie Mellon University, 1988.

[Laird *et al.*, 1987] J. E. Laird, A. Newell, and P. S. Rosenbloom. Soar: An architecture for general intelligence. *Artificial Intelligence*, 33(3), 1987.

[Laird *et al.*, 1990] J.E. Laird, K. Swedlow, E. Altman, and C.B. Congdon. Soar 5 user's manual. Technical report, University of Michigan, 1990. In preparation.

[Mishkin and Appenzeller, 1987] M. Mishkin and T. Appenzeller. The anatomy of memory. *Scientific American*, June:80–89, 1987.

[Moran and Desimone, 1985] J. Moran and R. Desimone. Selective attention gates visual processing in the extrastriate cortex. *Science*, 229:782–784, 1985.

[Newell, 1990] A. Newell. *Unified Theories of Cognition.* Harvard University Press, Cambridge, MA, 1990. (in press).

[Sperling, 1960] G. Sperling. The information available in brief visual presentations. *Psychological Monographs*, 74, 1960.

[Treisman and Gelade, 1980] A. Treisman and G. Gelade. A feature integration theory of attention. *Cognitive Psychology*, 12:97–136, 1980.

[Treisman and Schmidt, 1982] A. Treisman and H. Schmidt. Illusory conjunctions in the perception of objects. *Cognitive Psychology*, 14:107–141, 1982.

[Treisman, 1987] A. Treisman. Properties, parts, and objects. In K. R. Boff, L. Kaufman, and J. P. Thomas, editors, *The Handbook of perception and human performance.* Wiley-Interscience, New York, 1987.

[Wiesmeyer and Laird, 1990] M.D. Wiesmeyer and J.E. Laird. A computer model of visual search. Technical report, Artificial Intelligence Laboratory, The University of Michigan, March 1990.

On the Evolution of a Visual Percept

John Theios and Stephen T. Morgan
Department of Psychology
University of Wisconsin, Madison, WI 53706

Abstract. Human processing systems face a continual challenge of extracting only the most important information from the environment, resulting in awareness of some but not all of the available information. The current study investigates the psychophysical determinants of awareness. It examines the hypothesis that relative energy level of a stimulus is the critical factor in determining what comes to consciousness. In Experiments 1 and 3 two conceptually incompatible stimuli (one lexical, one pictorial) are presented successively in the same position of a tachistoscope for only 1 msec each, with a zero interstimulus interval. Observers report seeing one or the other, or nothing at all, at a more-or-less chance level when the stimulus durations are equal. As soon as one stimulus is given as little as one-quarter to one-half a millisecond more duration than the other, the longer stimulus is reported on 68% to 84% of the test trials. In Experiment 2 an advantage for word perception at these low energy levels was investigated and measured. The results of these experiments indicate that extremely small differences in duration (about .25 or .50 msec) were sufficient to bring one concept to the consciousness of the observer at the expense of the other. Small stimulus intensity differences were investigated in Experiment 4, yielding similar results. The results can be accounted for by contemporary parallel-distributed-processing, connectionist network models of perception and cognition which use a winner-take-all decision rule.

Experiment 1: The Complete Time Course of the Evolution of a Percept

Humans, as information processors, are faced with the challenge of continually gleaning information from the world. As the environment offers an infinite amount of information, the task of the processing system is to sufficiently and efficiently extract only what is most important. In this regard, we experience awareness of any particular stimulus only at the expense of other, competing stimuli. An obvious question to arise is "What are the determinants of awareness?" What is it that causes some information to rise to consciousness, while other information does not? The purpose of this paper is to investigate this issue. The experimental paradigm used is to present two conflicting stimuli within the same unit of psychological time, and manipulate the variable or variables which determine the resolution of the conflict. In short, we wish to discover the variables which are critical in determining the evolution of a percept from sensory stimulation to conscious cognition.

Connectionist Network Models with a Winner-Take-all Decision Rule. Parallel-distributed-processing, connectionist models provide a framework of conceptualizing visual processing in which information is represented within a network of interactive nodes. When two conceptually incompatible stimuli are presented in the same small unit of psychological time, it is likely that the two concepts may not reach consciousness simultaneously since consciousness requires the participation of the entire system. If multiple concepts are presented to an observer at a single point in time, the outcome may be likened to a

massively parallel race through the lower-order levels to a limited capacity, high order, cognition or consciousness. What is proposed here is that when two conceptually incompatible stimuli are presented in the same perceptual moment, both stimuli will activate their low-level representations within the system. As these representations in turn activate higher levels, the processing capacity of the system becomes more and more limited. At some point there is a bottleneck in which only one concept may be represented. The "challenge" to the competing concepts is to get through the lower levels quickly in order to arrive first at the point of the bottleneck, and thus capture control of attention or consciousness. In this "race for awareness" what factor is critical? The hypothesis under consideration is that it is the energy of the stimuli which will determine the victor. And to the victor, goes the spoils - conscious awareness. As Bloch's law describes energy as intensity times duration, whichever stimulus is presented for a slightly longer time, or at a slightly higher intensity should have a higher probability of capturing the awareness of an observer.

Method. The stimuli were the concepts UP, DOWN, LEFT, and RIGHT. Each trial consisted of presenting both a word and arrows pointing in one of the three directions not represented by the word. Experimental conditions manipulated the target position (first or second), target type (picture, word), and target duration. The "target" was the stimulus whose duration was manipulated on any particular test sequence, and the "standard" was the stimulus which was held at a constant duration of one millisecond. Target duration was then raised in increments of .25 ms for each successive test presentation, while duration of the standard stimulus remained constant at 1 ms. Each trial sequence continued until the observer correctly reported the target three times in a row. On the first trial of each sequence the two stimulus concepts were presented for only .25 and 1 ms, respectively. The interstimulus interval for all trials was zero. The observer's task was merely to identify what was presented. If the observer reported not knowing, he was forced to guess. On successive test trials, the target duration was then raised in increments of .25 ms until the observer correctly identified the concept on three successive presentations.

Results. The results of this experiment may be seen in Figure 1. When the target is presented at .25 ms, and the standard is presented for 1 ms, observers never report the target. As the target duration is increased to .50, .75 and then to 1 ms, recognition of the target increases to proportions of .035, .138, and .392. When the target duration is 1.25 ms (only .25 ms longer than the duration of the standard), it can be seen that target recognition increases dramatically to a mean proportion of .677. When the target duration is 1.5 ms, half a millisecond longer than the standard, the probability of reporting the target is .84. Probability of reporting the target continues to increase steadily as duration increases, until it asymptotes at 1.0, perfect recognition, at a 2 ms difference. What we have here is a complete psychometric recognition function which goes from 0 to 1.00 in a stimulus difference domain covering only 2 ms, -1 to +1 ms. This is close to a step function.

Experiment 2: Measuring the Unconditional Word Advantage

This experiment explores the advantage of recognition for words over recognition for less familiar pictorial stimuli. The experimental paradigm of Experiment 1 was used, **mutatis mutandis**. By presenting words

and arrows at an equal duration on the first trial of each sequence, it was hoped to get an unconditional measure of recognition differences based on stimulus surface structure. Recognition would be relatively unaffected by any immediately preceding perceptual events or sequences of responses.

Method. Six observers were used. All aspects of Experiment 2 were identical to Experiment 1 except the manipulation of target duration. On the first trial in every test sequence in Experiment 2 both stimulus concepts were presented for 1 ms each. On successive trials, target duration was raised in increments of .5 ms. The standard remained at 1 ms throughout the entire test sequence.

Results. At a zero difference in stimulus duration between target and standard, observers report the target stimulus at a proportion of .28. The fact that this value is so low indicates that with equal durations, each stimulus is masking the other somewhat. When there is as little as a one-half a millisecond difference in the stimulus durations, observers report the stimulus with the greater duration at a proportion of .69. The proportion of reporting the longer duration stimulus rises as the target duration increases until it asymptotes at 1.0 between a 2.5 and 3 ms difference in stimulus durations. Figure 2 shows the effect of target modality. It can be seen that word identification has an advantage over analogical pattern recognition. At the zero difference condition, observers correctly reported the word target at a proportion of .44. This may be compared to a .13 proportion of correctly reporting the pictorial stimuli at the same equal exposure duration condition. This value is at the chance level of one correct guess out of eight possibilities. Thus, the word stimuli completely masked the pictorial stimuli when the two exposure durations were equal. The difference in recognizability of these two stimuli diminishes as target duration increases. The psychometric functions for both conditions reach an asymptote of near 100% target recognition at similar duration differences between target and standard, approximately 2 ms. Target position (order) had no effect.

Discussion. The results of these two experiments strongly suggest that the energy value of a stimulus plays a crucial role in the evolution of a visual percept. In these experiments we manipulated stimulus energy by varying exposure duration. It has been shown that an exposure duration difference of as little as .25 or .50 ms between two contiguous, successively presented stimuli is enough for the slightly longer stimulus to capture the observer's attention or awareness. Moreover, this effect has been demonstrated to hold true regardless of whether the longer stimulus is presented first or second, or whether it is a picture or a word.

Experiment 3: Psychophysical Method of Constant Stimuli

In Experiments 1 and 2 the psychophysical method of ascending limits was used. On any given test sequence, the target and standard stimuli remained in the tachistoscope for as many trials as needed until the observer consistently reported the target on three consecutive trials. It could be argued that observers may have been aggregating featural and semantic information from one trial to the next. This argument could be made in spite of the fact that on the early trials of a test sequence the observers were unaware of the target and unable to report it. If so, they could have been building up a mental prototype of what the target was before they recognized it. If this was the case, then the recognition threshold values (the points at which the observers switched

from seeing and reporting the standard to seeing and reporting the target) may be lower than the "true" objective threshold values. In Experiment 3, in order to make sure the observer could not aggregate information from one test trial to the enxt, we switched from an ascending method of limits procedure to the method of constant stimuli in which the test stimulus pairs were chosen at random on each trial. Here, information on successive test trials are independent, and any stimulus information, conscious or unconscious, the observer may have picked up on one test trial could not possibly be of systematic help on the next test trial.

Method. Four observers were used. All aspects of Experiment 3 were identical to Experiment 1 except that the psychophysical method of constant stimuli was used. The standard was presented for 1 ms, but the exposure duration of the target was variable, ranging from .5 ms to 2 ms, in steps of .25 ms. The interstimulus interval between the target and the standard was zero msec. Randomly, from test trial to test trial, the target could be presented first or the standard could be presented first. Also, randomly from trial to trial, the target could be a word or a pictorial analog. For any given test series of 48 trials, the target exposure duration was choosen at random, and then each of the 12 pairs of test stimuli were randomly (without replacement) tested at that target duration in both stimulus orders and stimulus modalities.

Results. The results of Experiment 3 using random presentation of stimulus pairs were almost exactly the same as the results of Experiment 1 using the ascending method of limits.

Experiment 4: Effect of Differential Intensity Contrast on Recognizability

In our discussion of the results we have talked about relative stimulus energy being the deciding factor in determining what the observer cognitively sees. However, in our first three experiments we have only varied exposure duration differences, knowing Block's Law of "Energy is equal to intensity times duration in an interval up to 100 msec." In Experiment 4 we reduce the standard stimulus intensity in half, but increase its duration to 2 ms, yielding a standard of the same subjective brightness as in the first three experiments. Then we vary the intensity of the 2 ms duration target from three small brightness units below the standard to three small brightness units above the standard. At each intensity difference we use the method of constant stimuli.

Method. Five observers were used in this study. Experiment 4 was similar to Experiment 3 except for constant exposure durations and the manipulation of stimulus intensity. The duration of both the standard and target were fixed at 2 ms each. The intensity of the standard was set at the midrange value of our tachistoscope, 8.4 cd/m^2 for all test trials. In Experiment 4 the target was presented at an intensity value of 4.2, 5.6, 8.4, 13.1, or 20.5 cd/m^2.

Results. The results of this experiment may be seen in Figure 3. When the target was presented at 4.2 cd/m^2, and the standard was presented for 8.4 cd/m^2, the observers never reported the target. As the target intensity is increased to 5.6 and then to 8.4 cd/m^2, recognition of the target increases to proportions of .167 and .555. Thus, observers are at a chance level in reporting the target when the duration and intensity of the target and standard are exactly equal. When the target intensity is 13.1 cd/m^2 (only one brightness unit above the intensity level of the standard), target recognition increases dramatically to a

mean proportion of .883. When the target intensity is increased to 20.5 cd/m^2, only two units brighter than the standard, the probability of reporting the target is almost perfect, .977. Again, what we have here is a complete psychometric recognition function which goes essentially from 0 to 1.00 in an intensity difference domain starting at 4.2 cd/m^2 and going to 20.4 cd/m^2 in only four equal brightness steps.

Discussion. By investigating processing in the 1 to 2 ms stimulus onset asynchrony range we have done away with encumbering phenomena such as central pattern masking which cloud the issue at hand. We believe we have investigated the psychophysical determinants of consciousness in a most fundamental sense. As a result of the present investigations, the claim can be made that at the most basic level it is stimulus **energy** which is the critical determining factor in perception. The longer, brighter, stronger, larger, and lower spatial frequency environmental events are those that tend to capture the awareness of an observer. These higher energy stimuli (both forwardly and backwardly) mask their temporally equivalent but weaker cohort stimuli. If there are multiple stimuli available in the same small perceptual moment, then an energy advantage of even the slightest degree may be enough to allow the evolution of any particular percept at the expense of the others in the same small psychological time frame. A theoretical microstructure can be given by any number of contemporary connectionist network models of human information processing. Low-level featural information extracted from the input stimulus is assumed to activate subsequent internal representations, which are of increasingly higher and higher orders. In this manner, featural information provided by a stimulus evolves into a state of a high-level conceptualization as it flows through the neural net, being augmented in some areas and meeting resistence or inhibition in others. As awareness of a concept is represented as a pattern of activation in the system at the highest level, enlisting the participation of the system as a whole, no two conceptually incompatible concepts may be fully activated simultaneously within one small psychological moment in time. The interpretation given to the present results is that the featural information from both our test stimuli is extracted from the persisting, doubly exposed visual icon. As information is sent on for higher-order analysis, processing resources become more limited. At some point there is a bottleneck (perhaps just before the point of consciousness) which allows an awareness of only one of the two presented concepts. Within this interpretation, the first stimulus to reach the bottleneck is the stimulus of which the observer becomes consciously aware. Therefore, the situation may be thought of as a struggle between the two stimuli to get to the point of the bottleneck first, and thereby capture attention and conscious awareness. What is the critical factor in determining the outcome of this sensory struggle? The present experiments have provided evidence that it is the **energy** contrast level of a stimulus, relative to the competition which can provide the "winning edge". What is remarkable is the tremendous influence very small energy differences seem to have. Small intensity differences and duration differences of less than a thousandth of a second can sway the outcome of this perceptual competition in either direction. In summary, give me a half a millisecond, and I will change your mind.

Figure 1. Target detection, Exp. 1.

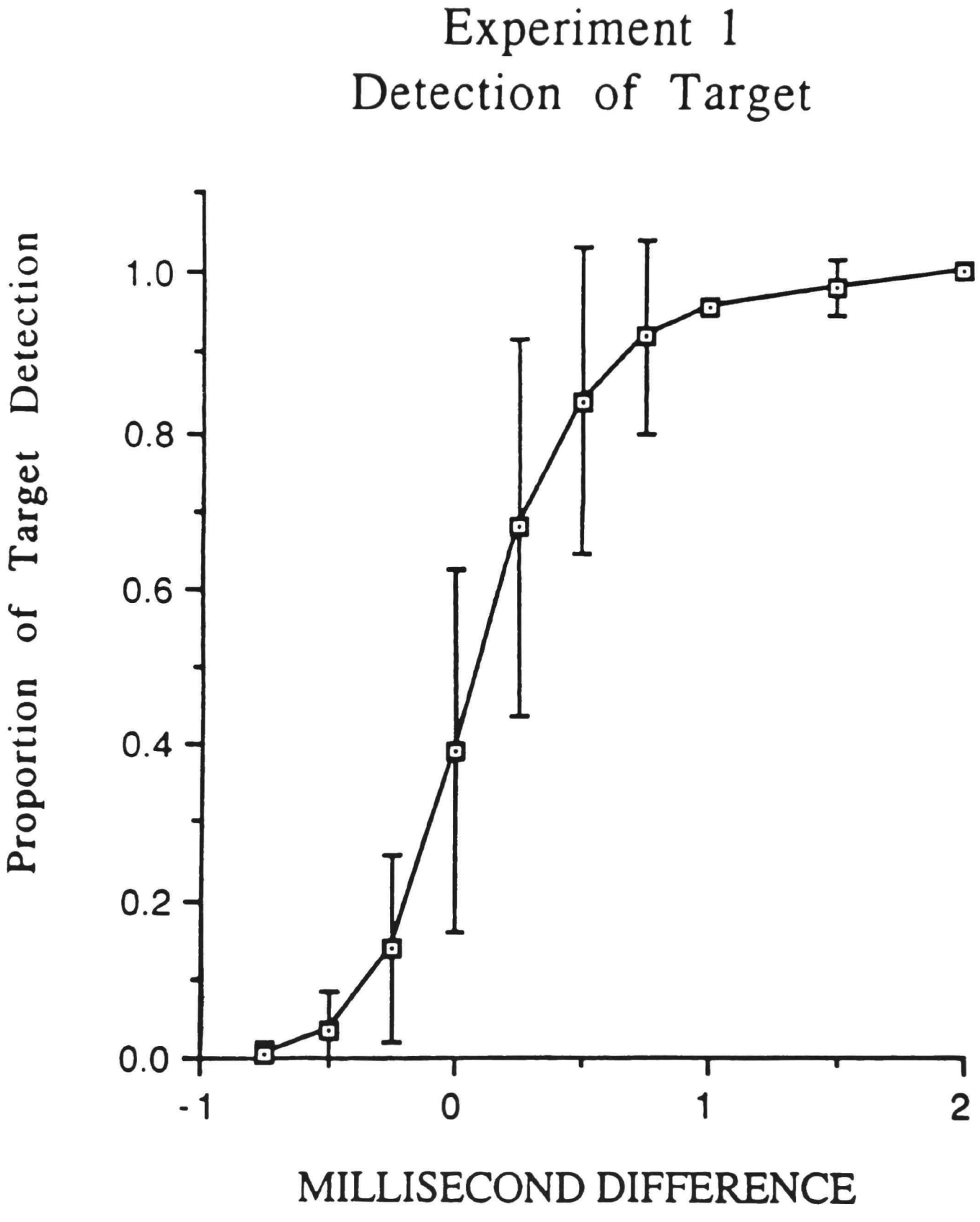


Figure 2. Lexical vs. Analogical recognition, Exp. 2.

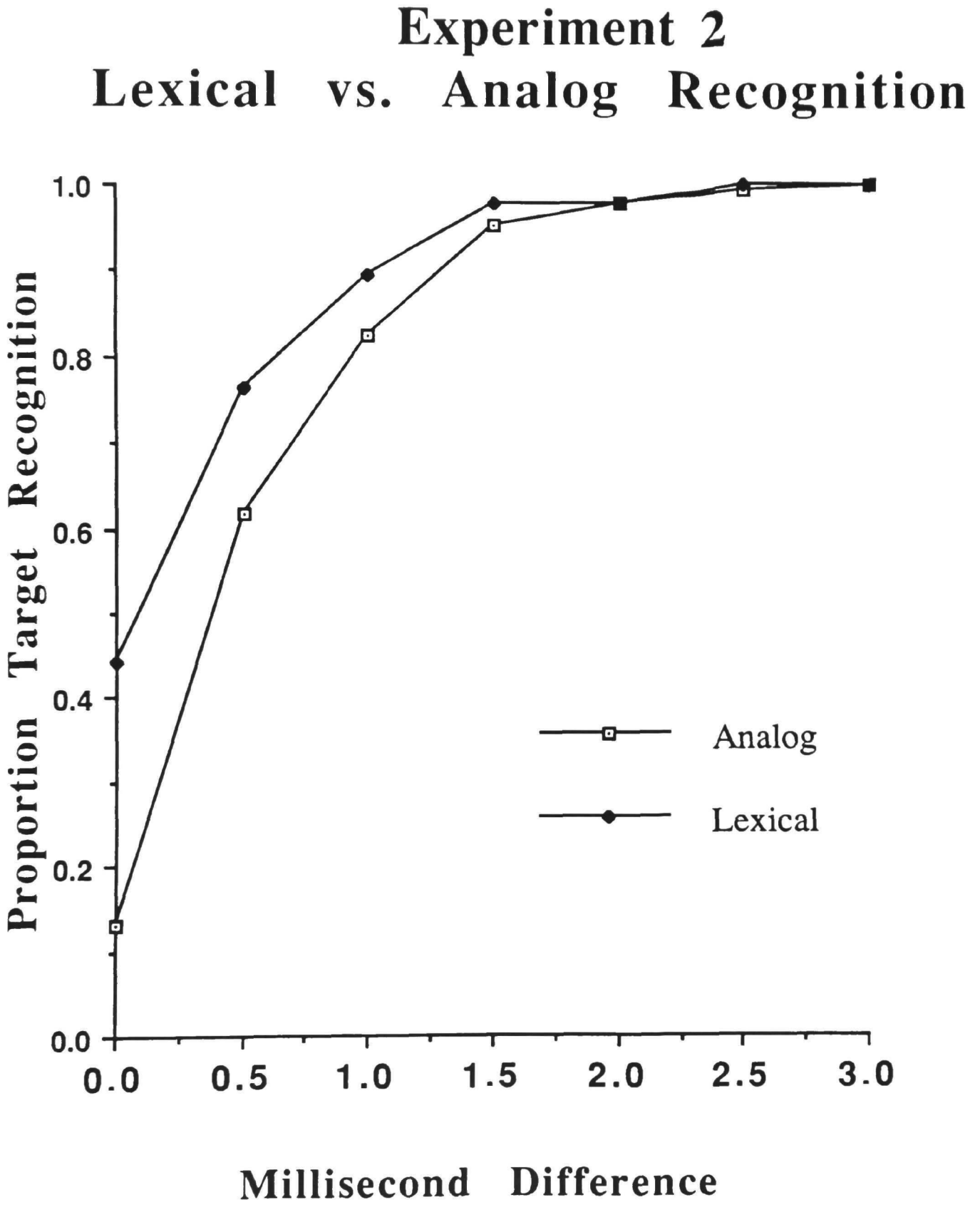
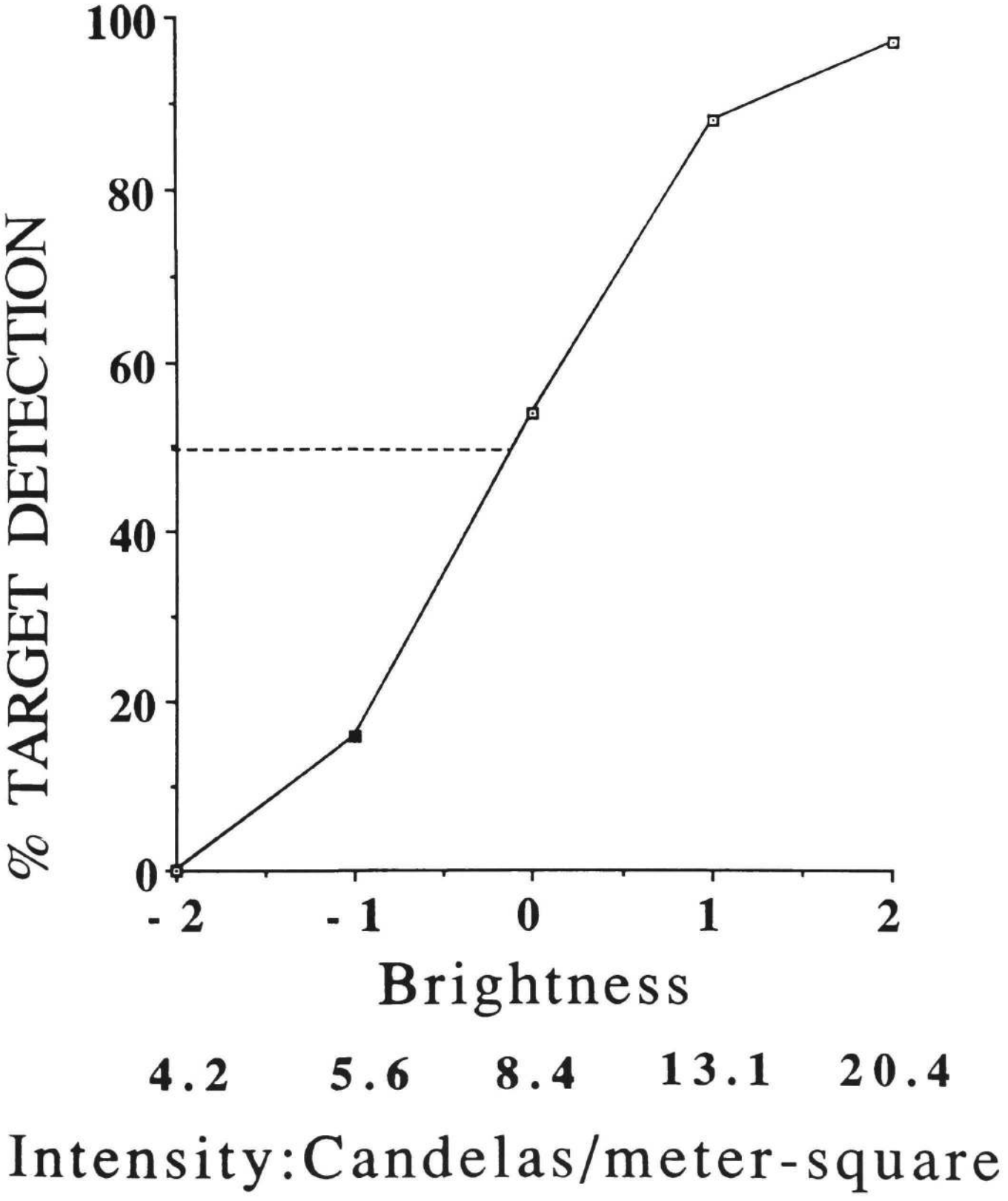


Figure 3. Target recognition, Exp. 4.



A Computational Model of Visual Pattern Discrimination in Toads*

DeLiang Wang and Michael A. Arbib

Center for Neural Engineering, University of Southern California
Los Angeles, CA 90089-2520, U.S.A.

ABSTRACT

It has been found behaviorally that visual habituation in toads exhibits locus specificity and partial stimulus specificity. Dishabituation among different configurations of visual worm stimuli forms an ordered hierarchy. This paper presents a computational model of the toad visual system involved in pattern discrimination, including retina, tectum, and anterior thalamus. In the model we propose that the toad discriminates visual objects based on temporal responses, and anterior thalamus has differingrepresentations of different stimulus configurations. This theory is developed through a large scale neural simulation. With a minimum number of hypotheses, we demonstrate that anterior thalamus in response to different worm stimuli shows the same hierarchy as shown in the behavioral experiment. The successful simulation allows us to provide an explanation of neural mechanisms for visual pattern discrimination. This theory predicts that retinal R2 cells play a primary role in the discrimination via tectal small pear cells (SP) while R3 cells refine the feature analysis by inhibition. The simulation also demonstrates that the retinal response to the trailing edge of a stimulus is as crucial for pattern discrimination as to the leading edge. New dishabituation hierarchies are predicted by shrinking stimulus size and reversing stimulus-background contrast.

1. Introduction

After repeated presentation of the same prey dummy in their visual field, toads and frogs may reduce the number of orienting responses toward the moving stimulus. This phenomenon is called *habituation*. Habituation has been extensively investigated, ranging from invertebrates, like *Aplysia* [1], where habituation seems to be independent of the specific patterning of the stimulus, to mammals where habituation exhibits stimulus specificity so that habituation to a certain stimulus pattern may be dishibituated by a different stimulus pattern [2]. Visual habituation in toads has the following characteristics [3]:

(1) *Locus specificity*. After habituation of an orienting response to a certain stimulus applied at a given location, the response can be released by the same stimulus applied at a different retinal locus.

(2) *Partial stimulus specificity*. Another stimulus given at the same locus may restore the response habituated by a previous stimulus. Only certain stimuli can dishabituate a previously habituated response. Experimental results [4] show that this dishabituation relation forms an ordered hierarchy of stimulus patterns (Figure 1), where only a pattern at a higher level can dishibituate the habituated responses of the stimuli at lower levels. We call this kind of stimulus specificity *partial* because the dishibituation relation is ordered rather than mutual.

The biological relevance of the stimulus-specific habituation phenomena may be to keep the IRM (innate releasing mechanism) for prey catching alert to "new" stimuli. However, the dishabituation hierarchy suggests that it is *configurational cues* of the stimulus and not only its "newness" which decide the toad's response [4]. It is reasonable to assume that toads have not developed the advanced spatial shape recognition capability of higher animals, but have developed the ability to recognize certain stimulus configurations, which, for example, are used in discriminating prey and predator.

Although Ewert and Kehl [4] demonstrated the behavioral responses leading to the hierarchy, they did not investigate the neural mechanisms involved, which must involve visual centers such as retina and tectum. For toads to exhibit the dishabituation hierarchy, there have to be differing representations of different stimulus shapes somewhere in their visual system. Unfortunately, physiological studies provide very little data on the response of these centers to a variety of stimulus shapes (for a review see [5]). In the Lara-Arbib model for this stimulus-specific habituation behavior [6], the discrimination of the stimuli in Fig.1 is made by retinal ganglion cell R2. They introduce a group of *ad hoc* functions each of which is used to emulate how a specific stimulus traverses the

* The research described in this paper was supported in part by grant no. 1RO1 NS 24926 from the NIH (M.A.Arbib, PI).

excitatory receptive field (ERF) of R2, but they did not give any biological justification for that *ad hoc* group of functions.

We develop a model for discriminating different worm-like stimuli which is able to demonstrate a class of cells whose firing rate in response to the different stimulus types exhibits the same order as shown in the dishabituation hierarchy. We hypothesize that these cells lie in the anterior thalamus, and thus suggest new physiological experiments to test our theory.

2. Distributed vs. Temporal: Basic Hypothesis

An object can be naturally coded by distributed activity in a group of neurons, or at some high level by firing activities of single cells. Here the former is denoted as *distributed coding* and the latter as *temporal coding*. Distributed coding is strongly favored by theoreticians due to considerations of reliability, although it seems that both are used in the object representation of primates. Do anurans represent various worm-like stimuli by distributed coding or temporal coding? Our basic hypothesis is that anurans represent objects by *temporal coding*. More specifically, the firing rate of neurons in some neural center of the toad visual system is higher in response to a stimulus in the upper part of the hierarchy than one in the lower part.

Generally speaking, distributed representation corresponds to neural associative memory, where the difference between two patterns is measured by their Hamming distance. However, the distributed coding fails to explain the partial stimulus-specificity since no ordering can be embodied in associative memory. This might be the case in higher animals like mammals where dishabituation could be accounted for by a comparator model [2], but this contradicts the observed ordered hierarchy in toads [4]. Moreover, the discrimination capability of toads is rather limited. In their original experiment, Ewert and Kehl did not find shapes other than those in Fig.1 that could be discriminated (Ewert, personal communication 1989). This limitation seems straightforward based on our hypothesis because the frequency spectrum can only be markedly differentiated into a number of levels and therefore the capacity is severely limited compared to distributed coding. It could be that amphibians, a phylogenetically older species than mammals, have not yet achieved the advanced distributed coding which has immense potentials in terms of capacity. Looked from the other direction, however, amphibians do reach the partial stimulus-specificity which does not seem to be obtained in invertebrates [1].

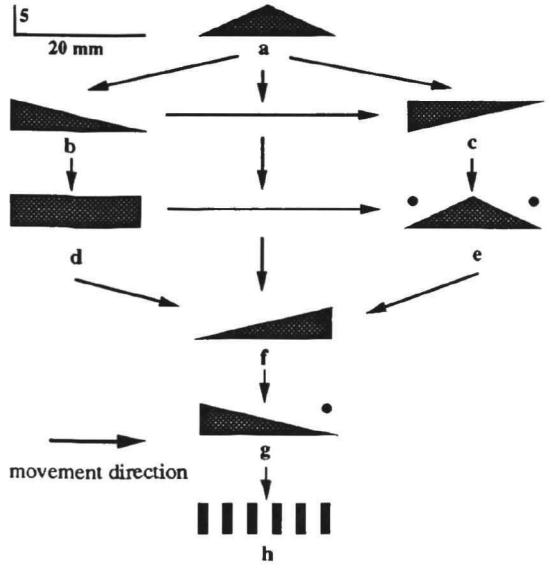


Figure 1. Dishabituation hierarchy for worm stimuli used in stimulus-specific habituation of toads. One stimulus can dishabituate all the stimuli below it. On the same level the left stimulus can slightly dishibituate the right one (from [4]).

A major criticism of temporal representation is that the reliability of this coding paradigm depends on single cells which are vulnerable in the nervous system. However, this criticism can be answered through redundancy of a cell population, as is actually the case in the nervous system. Neighboring neurons have almost the same receptive field, so the malfunction of a few cells would not matter so much because their neighbors could easily replace their role. If a localized neuron population is destroyed, we should expect localized malfunction of the temporal represention. In the anuran visual system, a localized scotoma from a localized lesion is found in various visual centers starting from retina. This suggests that visual information is represented in anurans by a temporal paradigm, since in classic models of distributed coding (for example see [7]) localized cell loss seems impairing only general associative abilities.

A direct prediction of our basic hypothesis is that the ordered dishabituation hierarchy is underlain by the different firing rates of certain neurons in the toad visual system. This prediction will be tested in simulations presented in the following sections. In the experiment of Ewert and Kehl [4], all stimuli are *20mm* long and *5mm* high. With a *70mm* distance of presentation from the toads, each stimulus is about *16°* long and *4°* high. The dot used is *1 mm* in diameter which is about *1°*.

The locus specificity that toads and frogs show for dishabituation conforms well to temporal coding. *Retinotopy*, one form of locus specificity, has been extensively observed both in tectum and in thalamus of toads [3]. Although the adaptive significance for anurans treating the same stimulus at different locations as different ones

is not finally resolved, it does demand extra memory capacity compared to a memory scheme which achieves locus-constancy, but toads need an accurate location memory.

3. Retinal Coding

An elementary model of the retinal receptive field uses two mechanisms in deciding retinal responses: (1) an excitatory center and (2) an inhibitory surround. Both mechanisms are described by spatially Gaussian-distributed curves around a common midpoint, and the whole neuronal response is formed as a difference of Gaussians (DOG). Our retinal analysis is mainly based on the Teeters' retina model [8] since it provides the most detailed account of the toad retina to date. The model for the anuran retina prior to ganglion cells consists of four cascade layers: *receptor*, *horizental*, *biopolar* and *amacrine*. Three different types of ganglion cell were identified in the retinotectal projection of toads which correspond to R2, R3 and R4 in frogs. The response of R2 and R3 to three classes of stimuli used in the Ewert laboratory is summarized in Fig.2 [9]. Note that only the response to the leading edge is shown in the recording. In the model, each cell type corresponds to a two dimensional matrix, with a single cell identified by $m(i,j,t)$ representing the membrane potential of the neuron at position (i, j) at time t. $I(i,j,t)$, the input to the neuron at position (i, j), is created by a convolution of a kernel which approximates a DOG with the corresponding output from a previous cell matrix

$$I(i,j,t) = (k*S)(i,j,t) \tag{1}$$

where $*$ represents convolution, S indicates the output of the previous layer, and a kernel element $k(x,y)$ is defined as

$$k(x,y) = \begin{cases} W_e \exp[-(x^2+y^2)/(2\sigma_e^2)] - W_i \exp[-(x^2+y^2)/(2\sigma_i^2)] & \text{if } x^2+y^2 \le R^2 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where R is the radius of a receptive field measured in degrees of visual angle, W_e and W_i combine together to determine the weight from the previous layer to the current one. The Mexican hat of the kernel, the activity distribution of a receptive field, is uniquely determined by parameters W_e, W_i, σ_e, and σ_i.

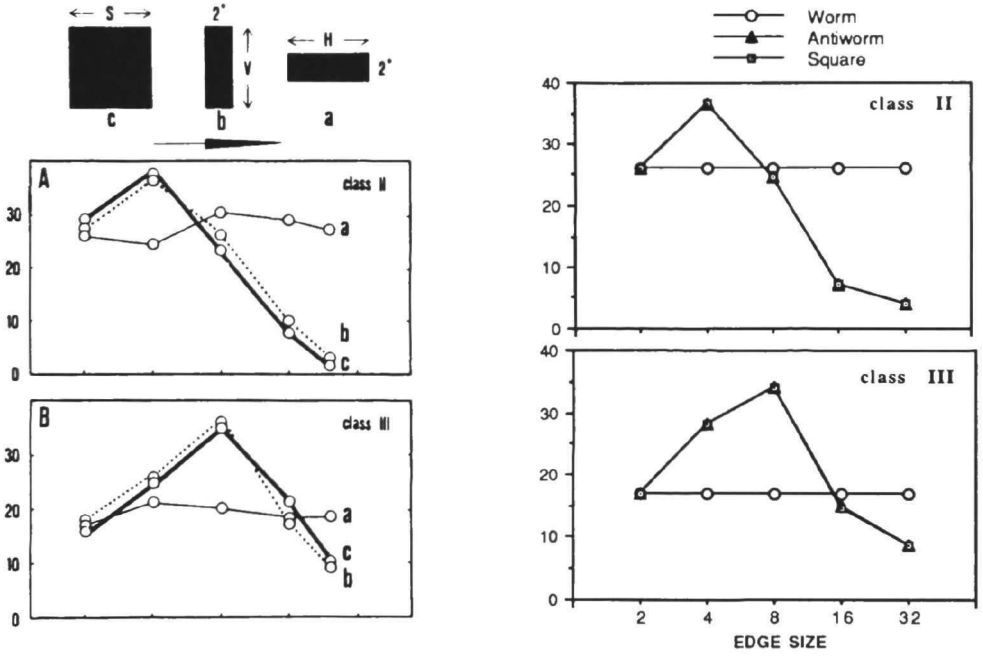


Figure 2. The model response of R2 and R3 to worm, antiworm, and square. **left**: the experimental data [9]. **right**: Response of our modified retina model. Each point in the figure represents the temporal average firing rate in response to the corresponding stimulus.

The retina model in this paper is a modified version of the Teeters model which more closely simulates the biological data. Simulation results from our modified model are presented in Fig.2 for the two types of ganglion cells respectively, together with the biological data. The average firing rate of a neuron is computed by the temporal integration of its instantaneous firing rate divided by the time period during which a non-zero firing rate is consecutively elicited. The response of R4 is neglected due to its insensitivity to difference between the worm stimuli. Our simulation result has demonstrated that without a significant trailing edge response, it is impossible to discriminate the different worm configurations even at the retina level.

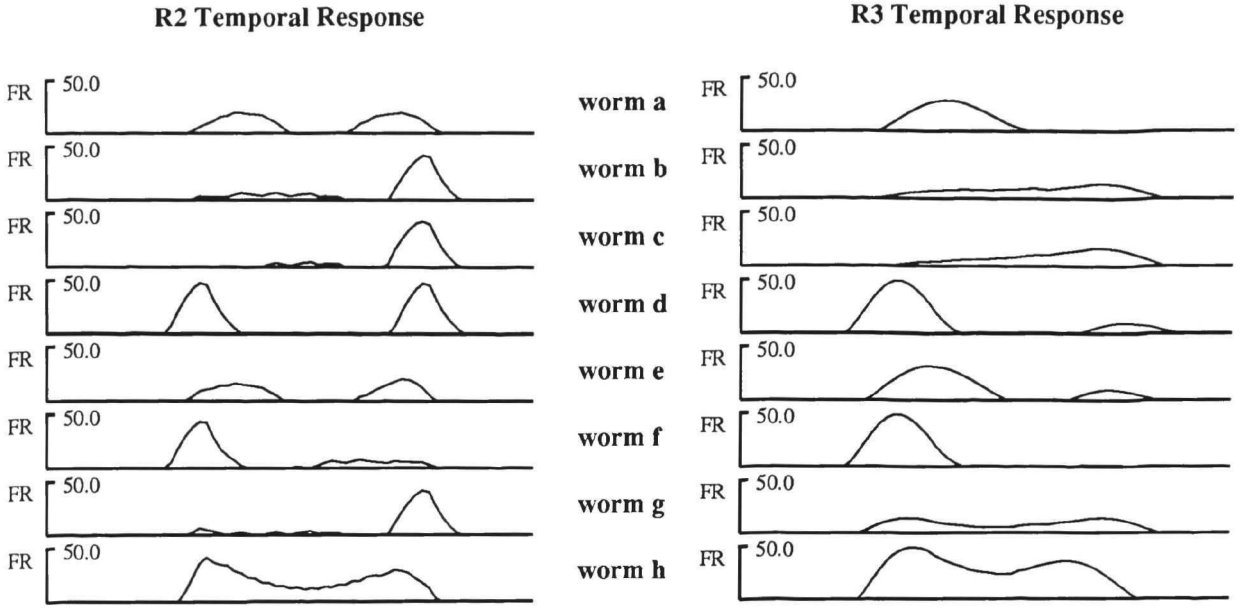


Figure 3. The temporal firing rate of R2 (**left**) and R3 (**right**) to the *8* worm-like stimuli from the retina model. Time runs from left to right, and all the stimuli are moving from left to right. The unit of the numbers in the figure is impulses per second.

Tsai and Ewert [10] recently found that R2 cells show no preference in response to both edges of a stimulus if neuronal adaptation is not taken into account, while R3 cells show a much stronger off-channel (from white to black) than on-channel (from black to white) response, which correlates with the behavior. In this simulation, the R3 response to trailing edge is modeled with a *1.0/0.2*, and the R2 response with a *1.0/1.0* ratio of off- to on-channel contribution. In particular, the R2 membrane potential is described by

$$\tau_{r2} \frac{dm_{r2}(i,j,t)}{dt} = -m_{r2}(i,j,t) + (k_{r2} * (S_{ath}+S_{atd}))(i,j,t) \qquad (3)$$

where τ is the time constant of the above leaky integrator model. Subscripts indicate the neuron types, e.g., k_{r2} stands for the kernel of a R2 cell as defined in (2). The detailed definition of S_{ath} and S_{atd} is given in [8].

Fig.3 shows the temporal responses of a R2 neuron and a R3 neuron to the 8 worm-like stimuli in Fig.1 from our modified retina model. Here only firing rate is displayed, which equals $m(t)$ if $m(t) > 0$ and 0 otherwise for both R2 and R3 cells. In terms of single cell response, a vertical bar of *4* ° height elicits the strongest response in both R2 and R3 cells, and the more inclined is a stimulus edge, the less efficient is it to trigger a retinal response. This is because the more inclined is a stimulus edge with the same vertical height, the larger does it encroach on the inhibitory surround and longer is the duration of the response. Note the R2 effect of the dots in worm **e** relative to **a** and worm **g** relative to **b** in Fig.3. Since the dot is encroaching R2's IRF while the edges of stimuli **e** and **g** traverse R2's ERF, worms **e** and **g** elicit smaller R2 responses than worms **a** and **b** respectively. Note also that the distance between the two peaks for R2 leading/trailing response corresponds to the distance between the middle points of the leading edge and the trailing edge of the stimulus. In the simulation, the density of receptors is *1 cell / 0.5 degree* while the ganglion cell density is *1 cell / 2 degree* resulting in a *4 to 1* density ratio. Each ganglion cell has a receptive field of approximately *20x20* degrees or *40x40* receptors.

4. Tectal Relay

Based on anatomical data and functional lesion data, Ewert [11] suggested that the basic pathway for habituation is: *retina* → *tectum* → *aTH* (anterior thalamus) → *MP* (medial pallium) → *tectum*. This pathway is refered to as *loop(2)* and is generally supposed to be responsible for modulation of the innate releasing behaviors of amphibians. In this paper, we are only concerned with the first part of this loop: *retina* → *tectum* → *aTH*, where the discrimination of the stimuli is presumably achieved.

In this model, the optic tectum discriminates prey from predator, but we hypothesize that it does not play a major role in the finer pattern discrimination that underlies the dishabituation hierarchy, but rather relays the input from retina to anterior thalamus where the visual information is further processed and carried up to telencephalon. As for stimulus-specific habituation, behavioral data show that the releasing values for all the stimuli in the hierarchy are almost the same [4]. Also, the prey-catching behavior shows off-channel preference, which correlates very well with the neuronal activities in R3 and T5(2) cells. This finding leads Tsai and Ewert [10] to propose that R3, not R2, carries the primary information to the prey analysis circuitry located in tectum. However, as pointed out in the previous section, the response to the trailing edge should have a significant role in worm discrimination. This suggests that R2, so called "net convexity detectors", may be more involved in the discrimination of worm-like stimuli than R3. This trailing edge consideration tends to downplay the tectum as a major neural center on "subworm" discrimination.

Lázár et.al. [12] found that in anurans the main projection units to the anterior diencephalon from tectum are the small piriform neurons (SP), which are located in layer 8 of the tectum. This important finding leads us to assume that SP cells relay the visual information concerning worm discrimination. In the present model SP cells receive R2 inputs, which is consistent with the tectum model of Cervantes et al. [13], and projects to anterior thalamus. Since R2 projects to SP topographically and the SP's role in this model is to relay R2 activity, to simplify the implementation the neuronal response of SP is made equal to that of R2 to the worm stimuli.

5. Integration in Anterior Thalamus

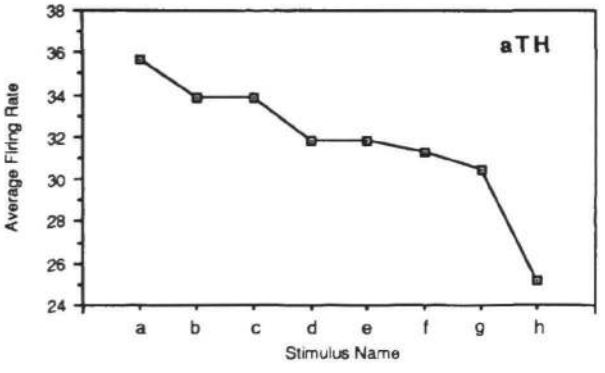


Figure 4. AT response to the *8* worm-like stimuli shown in Figure 1. The *8* average firing rates are ordered, which corresponds to the ordered hierarchy in Figure 1. In the simulation, $\tau_{at} = 0.065$, $\theta_{at} = 13.0$, $W^{e}_{sp} = 0.0091$, $W^{i}_{sp} = -0.003$, $W^{e}_{r3} = 0.0095$, $W^{e}_{r3} = -0.003$, $m_1 = n_1 = 6$, and $m_2 = n_2 = 12$.

The anterior thalamus receives ascending R3 retinal projections [5] and SP tectal projections [12]. Among other ascending projections to telencephalon, aTH has a direct projection to the medial pallium, where visual information is presumably stored. Compared to the optic tectum and the caudal thalamus, aTH is much less understood in terms of neurophysiology and morphology. Functionally, it has been suggested that aTH forms part of the anatomical substrate by which visual information reaches the medial pallium [14], and it was found that large ablations of aTH usually depressed the prey-catching behavior [15]. Also aTH has been proposed as part of the modulatory loop(2) [11]. However, the kind of visual processing performed by aTH remains unknown.

We offer in the present model a definite hypothesis: Based on the special position of aTH and our previous analysis of visual information processing, we propose that it is the anterior thalamus where the pattern discrimination is finally achieved by neuronal responses. In this model, AT neurons receive excitatory-center inhibitory-surround inputs from tectal SP cells, and direct inhibitory inputs from R3 cells. Quantitatively,

$$\tau_{at} \frac{dm_{at}(i,j,t)}{dt} = -m_{at}(i,j,t) + (k_{at1} * S_{sp})(i,j,t) - Max\,[0, (k_{at2} * S_{r3})(i,j,t)] \qquad (4)$$

where $k_{at1}(x,y) = W^{e}_{sp}$, if $|x| \le m_1$ and $|y| \le m_1$; W^{i}_{sp}, if $m_1 < |x| \le m_2$ and $m_1 < |y| \le m_2$; and *0* otherwise. $k_{at2}(x,y) = W^{e}_{r3}$, if $|x| \le n_1$ and $|y| \le n_1$; W^{i}_{r3}, if $n_1 < |x| \le n_2$ and $n_1 < |y| \le n_2$; and *0* otherwise. $S_x(t)$, $x \in \{sp, r3\}$, represents the firing rate of neuron type x, and $S_{at}(t) = m_{at}(t) - \theta_{at}$, *if* $m(t) \ge \theta_{at}$ and *0* otherwise.

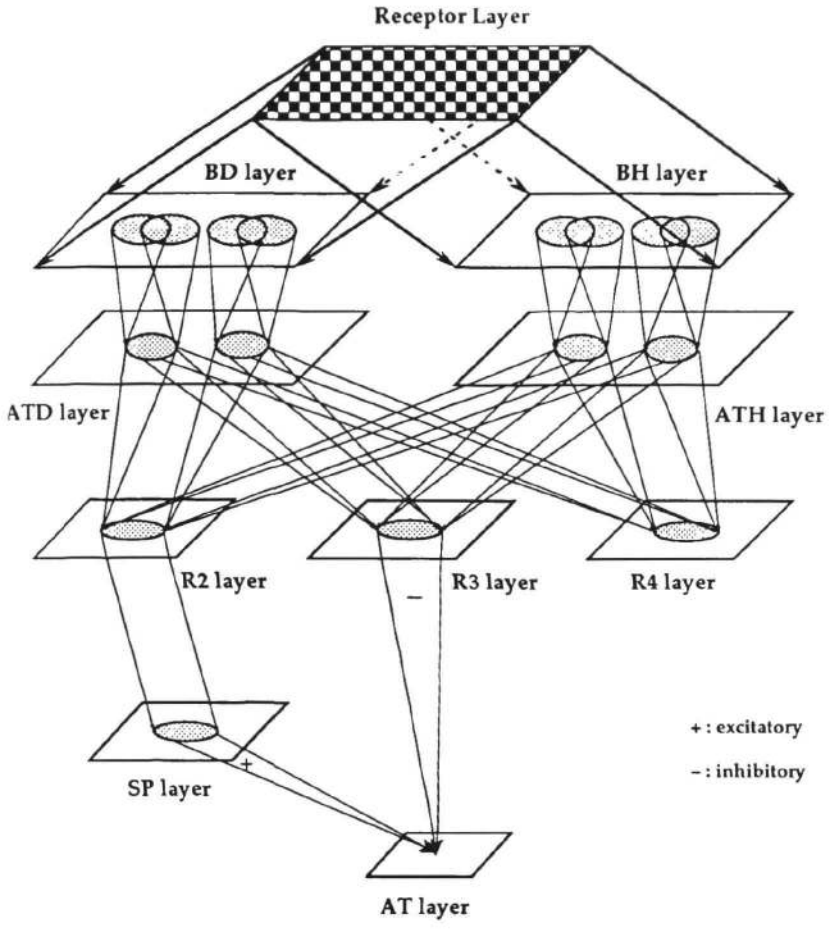


Figure 5. Diagram of the entire model used in this simulation project. Retina, tectum and anterior thalamus have been incorporated in the model. For explanation see text.

Figure 4 shows the average firing rates of a single AT neuron to the 8 worm-like stimuli. See the legend for the values of the parameters introduced above. The result clearly matches the ordered dishabituation hierarchy in Fig.1. Not only do stimuli higher in the hierarchy generate larger AT responses, but the stimulus pairs **b**–**c** and **d**–**e** which are on the same level in the hierarchy generate nearly equal responses. The modeled dishabituation hierarchy is the same as in Fig.1 except that the model cannot create the preference of stimulus **b** over **c**, which is weakly exhibited in the animal.

In summary, we propose the following mechanisms to explain the dishabituation hierarchy in Fig.1.

(1) Both the leading edge and the trailing edge of a worm stimulus have to be taken into consideration.

(2) The receptive field of AT neurons is big enough to "see" both the leading and the trailing edge. Stimulus **a** elicits the biggest response, particularly bigger than stimulus **d**, because both diagonal edges elicit strong responses in R2 cells (see Fig.3) and these responses can be best integrated in AT cells due to the small distance between the midpoints of its leading and trailing edge response.

(3) Stimuli **b** and **c** are prefered to stimulus **f** because the inhibition of R3 cells, which has off-channel preference, is bigger for **f** than for **b** and **c**.

(4) Stimuli with dots appear lower in the hierarchy because they elicit smaller R2 response due to IRF interaction.

(5) A striped pattern elicits the smallest response in AT neurons because of R3 inhibition.

6. Conclusion and Predictions

Inspired by the behavioral results [4] resulting in the dishabituation hierarchy, the present model represents an integration of behavioral, physiological, anatomical, and theoretical studies on the brain in order to postulate mechanisms for pattern discrimination in amphibians.The model is tested by a large-scale computer simulation which incorporates retina, tectum and anterior thalamus. The anatomy of the model is summarized in Fig.5. In the figure, conical projections represent on-center off-surround convergence, while the cylindrical projection from the R2 layer to the SP layer represents a 1 to 1 mapping. The connections from the receptor layer to both the BD and BH layer also constitute a small many-to-one convergence. The receptor layer contains *140x140* cells which correpond to *70°x70°* visual field. Bipolar and amacrine cell layers consist of *140x140* cells respectively which correspond to the receptor layer. Three types of ganglion cells have been modeled, each consisting of *25x25* cells which correspond to *70°x70°* visual field since the ganglion cells have *20°* RF and lie *2°* apart. The R2 layer projects to the SP layer in the tectum, and the SP layer and R3 layer together converge on the AT layer in the anterior thalamus, where the

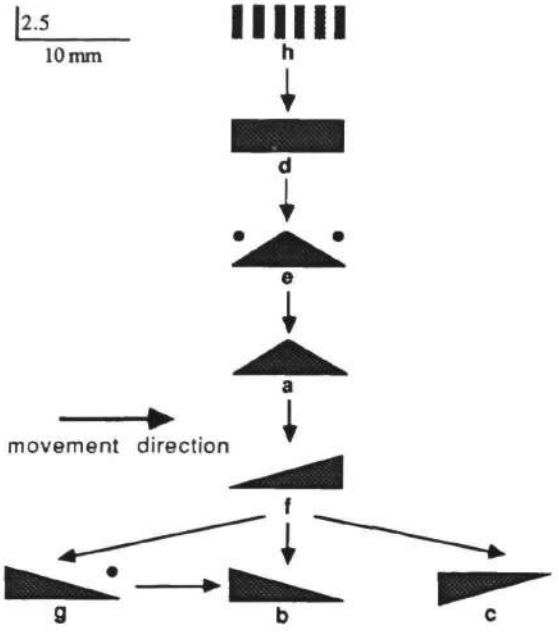


Figure 6. Dishabituation hierarchy predicted from this model by shrinking stimulus size. All the stimuli are *10 mm* long and *2.5 mm* high. The same set of stimulus configurations is used as in Figure 1.

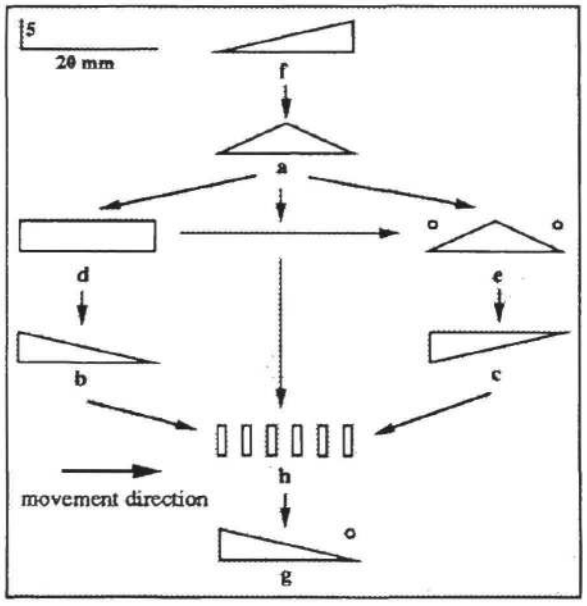


Figure 7. Dishabituation hierarchy predicted from this model by reversing contrast direction. In contrast to Fig.1, white stimulus is moving against black background. The same set of stimulus configurations is used as in Figure 1.

worm-like pattern discrimination is finally achieved. The entire simulation contains about *100,000* cells. Bitmap stimuli are used.

The simulation of the pattern discrimination was done after the retina model was fixed. Since the retinal responses are constrained strongly by the experimental data [9, 10], the retinal model, even its various parameters, cannot be modified to fit other simulation purposes since otherwise the original match between the model and the data could not be preserved. This represents a real challenge for later simulations based on retinal output. On the other hand, this requirement also provides a strict testbed for hypotheses and neural models.

A major postulate of this paper is that toads represent objects by temporal coding. This theory explains the locus-specificity of habituation and the dishabituation hierarchy in the following way: a stimulus at a higher position of the hierarchy elicits a stronger response at some location than that elicited by a stimulus at a lower position. Based on the theory, we have proposed a concrete neural model which successfully simulates the experimental data. The following list provides a number of model predictions:

(1) When the animal is presented with the different worm-like stimuli, they will elicit different neuronal responses at a certain neural center, and the order exhibited based on average firing rate corresponds to the order exhibited in the dishabituation hierarchy.

(2) Retinal ganglion cell type R2 plays a primary role in the discrimination of the stimuli, since R2 responds best to small moving objects and detects equally well both the leading and trailing edge of a stimulus.

(3) In the discrimination of different "sub-worms", the optic tectum serves only to relay information from retina to aTH via SP cells.

(4) R3 cells have an inhibitory role in worm pattern discrimination. This is due to their off-channel preference (from white to black).

(5) Anterior thalamus is the structure which reflects the final pattern discrimination. This structure receives excitatory projections from SP and inhibitory projections from R3.

In terms of stimulus size, the current model will create different hierarchies based on different sizes of worm-like stimuli. After completing the previous simulations, we shrinked the size of all the stimuli to *10mm* long and *2.5mm* high correponding to *8°* by *2°*, and tested these stimuli. Fig.6 shows the dishabituation hierarchy predicted by this model. A remarkable difference has been revealed compared to Fig.1. Particularly, stimulus **h** lies at the top, in contrast to the bottom position in Fig.1, and stimuli with dots appear higher in the hierarchy, reversing the original relation exhibited in Figure 1. Our

explanation is that since the stimulus size is halfed compared to Fig.1, the previous IRF interaction in the R2 receptive field are converted into an ERF interaction which strengthens overall responses. This ERF interaction is particularly manifested by stimulus **h**. Note that the R3 inhibition in AT neurons is relatively smaller than the excitation from SP cells, and thus cannot prevent stimulus **h** from inducing a strong AT response. This model discovery leads us to postulate that the effect of dot and striped pattern is relative to stimulus size in pattern discrimination. Furthermore in terms of multiple stimulus effect, this prediction suggests that if multiple stimuli lie close to each other they tend to cooperate to form a stronger response than any one of them, otherwise if the stimuli lie far from each other they tend to compete and conteract each other's response.

In this model of pattern discrimination both on-channel and off-channel effects are considered important. We have tested the same stimulus patterns as in Fig.1 but reversed the contrast-direction, i.e. white stimuli moving against a black background (w/b). A new dishabituation hierarchy is found in our simulation, as shown in Figure 7. The response of R2 cells with w/b is the same as with b/w, but now R3 cells show a trailing edge preference. These predictions have to be tested experimentally.

References

[1] Kandel E. (1976): *Cellular basis of behavior: an introduction to behavioral neurobiology.* Freeman: New York.

[2] Sokolov E. (1975): Neuronal mechanisms of the orienting reflex. In: *Neuronal mechanisms of the orienting reflex.* Sokolov E. and Vinogradova O. (eds), Lawrence Erlbaum/Hillsdale: New York.

[3] Ewert J.P. (1984): Tectal mechanisms that underlie prey-catching and avoidance behaviors in toads. In: *Comparative Neurology of the Optic Tectum,* Vanegas H. (ed). Plenum: New York, London.

[4] Ewert J.P. and Kehl W. (1978): Configurational prey-selection by individual experience in the toad *Bufo bufo. J.Comp.Physiol.* 126, 105-114.

[5] Grüsser O.J. and Grüsser-Cornehls U. (1976): Neurophysiology of the anuran visual system. In: *Frog Neurobiology.* Llinas R. and Precht W. (eds). Springer: Berlin, Heidelberg, New York.

[6] Lara R. and Arbib M.A. (1985): A model of the neural mechanisms responsible for pattern recognition and stimulus specific habituation in toads. *Biol.Cybern.* 51, 223-237.

[7] Hopfield J.J. (1982): Neural networks and physical systems with emergent collective computational abilities. *Proc.Nalt.Acad.Sci. USA* 79, 2554-2558.

[8] Teeters J.L. (1989): A simulation system for neural networks and model for the anuran retina. *Technical Report* 89-01, Center for Neural Engineering, University of Southern California, Los Angeles.

[9] Ewert J.P. (1976): The visual system of the toad: behavioral and physiological studies on a pattern recognition system. In: *The amphibian visual system: a multidisciplinary approach,* Fite K. (ed). Academic Press: New York.

[10] Tsai H.J. and Ewert J.P. (1987): Edge preference of retinal and tectal neurons in common toads (*Bufo bufo*) in response to worm-like moving stripes: the question of behaviorally relevant "position indicators". *J.Comp.Physiol.* A, 161, 295-304.

[11] Ewert J.P. (1986): Neuroethology: toward a functional analysis of stimulus-response mediating and modulating neural circuitries. In *Cognitive processes and spatial orientation in animal and man,* part 1, Ellen P. and Thinus-Blonc C. (eds). Dordrecht: Martinus Nijhoff.

[12] Lázár Gy., Toth P., Csink Gy., and Kicliter E. (1983): Morphology and location of tectal projection neuron in frogs: A study with HRP and Cobalt-filling. *J.Comp.Neurol.* 215, 108-120.

[13] Cervantes-Perez F., Lara R., and Arbib M.A. (1985): A neural model of interactions subserving prey-predator discrimination and size preference in anuran amphibians. *J.theor.Biol.*, 113, 117-152.

[14] Neary T.J. and Northcutt R. (1983): Nuclear organization of the bullfrog diencephalon. *J.Comp.Neurol.*, 213, 262-278.

[15] Ingle, D. (1980): The frog's detection of stationary objects following lesions of the pretectum. *Behav.Brain Res.*, 1, 139-163.

Binding and Type-Token Problems in Human Vision

Nancy G. Kanwisher
Department of Psychology
University of California, Berkeley
NGK@garnet.berkeley.edu

ABSTRACT

Two computational problems which are trivial for symbol-manipulating systems but which pose serious challenges to connectionist networks are the binding problem and the type-token problem. These difficulties arise because representations in connectionist networks do not automatically i) specify which features go with which object tokens, or ii) distinguish between different tokens of the same type. Nevertheless, these processing shortcomings may constitute advantages when connectionist networks are taken as models of human visual information processing. Perception research shows evidence not only of binding errors, for example in Treisman's illusory conjunctions (Treisman and Schmidt, 1982), but also of type-token errors, as seen in repetition blindness (Kanwisher, 1987) and other phenomena.

INTRODUCTION

There are a number of computational problems which symbol-manipulating systems can handle in a natural and straightforward way, but which pose a special challenge to connectionist networks (Pinker and Prince, 1988; Fodor and Pylyshyn, 1988; Norman, 1986). One of the most notorious of these problems has been variously referred to as the "binding problem" (Hinton, McClelland, and Rumelhart, 1986; Smolensky, 1987), the "indexing problem" (Feldman and Ballard, 1982), and the "tag assignment problem" (Strong and Whitehead, 1989). Roughly speaking, the binding problem refers to the fact that when several things are being represented simultaneously in a network, it is difficult to keep track of what information goes with what. For example, there may be no way to distinguish the visual representation of a red X and a green O from the representation of a red O and a green X.

It is not that the binding problem is unsolvable in connectionist models. Solutions for particular domains have been offered, for example, by Touretzky and Hinton (1986), Smolensky (1987), McClelland (1986), and Mozer (1987). The point is rather that binding is not automatically and naturally accomplished in connectionist models, as it is in symbol-manipulating systems. Thus, a good deal of extra effort is necessary to solve the binding problem in a connectionist model, often at the expense of some of the advantages of parallel processing.

Nevertheless, this processing shortcoming may constitute an advantage when connectionist networks are taken as models of human visual information processing. In particular, there is evidence that the human visual system may suffer from a binding problem of its own. If so, then connectionist networks might provide more accurate models of some stages of human vision than do symbol-manipulating models. Similar points have been made by Norman (1986), Rumelhart and Norman (1982), McClelland (1986), and Mozer (1987).

One example of a human perceptual limitation in binding is the illusory conjunction of visual features described by Treisman and Schmidt (1982). Previously implicated in this context by Hinton and Lang (1985) and Strong and Whitehead (1989), illusory conjunctions are incorrect recombinations of primitive visual features. Treisman and Schmidt showed that when visual attention is diverted from a display containing, for example, a red "X", a blue "T", and a green "O", subjects sometimes confidently reported seeing a green T" or a red "O". This and other findings have led Treisman and Gelade (1980) to hypothesize that early parallel stages of visual processing allow identification of features, but not binding of those features to their locations or to one

another. In Treisman's model the binding problem is solved at a subsequent serial stage in which visual attention acts as a gate to allow only one item to be processed at a time (see also Crick, 1984; Koch & Ullman, 1985, and Mozer, 1988; for evidence against temporal binding see McLean, Broadbent, & Broadbent, 1982; Keele, Cohen, Ivry, Liotto, & Yee, 1988).

Because it emphasizes the distinction between parallel processing of primitive features and serial processing of objects and locations, Treisman's model deals most naturally with illusory conjunctions of primitive features in spatial arrays. However, illusory conjunctions also happen for more complex visual categories like words, and affect stimuli displayed in serial lists as well as spatial arrays. For example, Lawrence (1971) showed subjects word lists displayed in rapid serial visual presentation (RSVP). When subjects were asked simply to report the single upper case word in an RSVP list of lower case words, they often reported not the actual capitalized word, but the word that followed it in the sequence. Temporal illusory conjunctions have also been shown for a picture and its frame (Intraub, 1989), and for letters and their colors (McLean, Broadbent, and Broadbent, 1982). Thus, insofar as illusory conjunctions are indicative of a binding problem in visual processing, the problem does not seem to be restricted to the early stages which extract primitive visual features.

A related problem of connectionist networks is the difficulty of representing multiple occurrences of the same category, or multiple "tokens" of the same "type" (Norman, 1986; McClelland, 1986). For spatial arrays, this could be thought of as a subset of the binding problem, since one obvious way of representing the fact that there are two of something would be to bind it to two different locations. This argument has been made by Mozer (1989), who has shown that people often judge a letter array to have fewer items if the array contains repeated letters than if it does not. He argues that the difficulty with repetitions arises because "given the loss of location information, difficulty in detecting repetitions of an object seems certain" (Mozer, 1989, pg. 298). In other words, according to Mozer, the type-token problem can be seen as a *result* of the (more general) binding problem. A similar argument (Keren and Boer, 1985) has been offered to account for the repeated letter inferiority effect (Bjork and Murray, 1977; Santee and Egeth, 1980) in terms of spatial uncertainty.

I will argue here that the human visual system indeed has a serious type-token problem, but that it cannot be accounted for as a simple case of the already-established binding problem. Rather, there seems to be some extra difficulty in encoding two tokens of the same type, over and above the difficulty of binding two nonidentical items to their locations in space or time. To make this point, I will describe some research on a recently-discovered visual phenomenon called "repetition blindness" (Kanwisher, 1987).

Repetition blindness (RB) was first discovered in a pilot observation of Intraub and Potter (Potter, personal communication, 1985), in which subjects were asked to say which picture occurred twice in an RSVP sequence of pictures. At the rapid presentation rates used (about 7 items/second), viewers generally had the impression that they saw most of the pictures but that none of them occurred twice. Further exploration of repetition blindness for words in RSVP (Kanwisher, 1986) showed that the difficulty of detecting repeated words happened even when three words intervened between the two occurrences, and even when the two occurrences differed physically (i.e., one was in upper case and one in lower case).

Repetition blindness was demonstrated most strikingly, however, when repeated words were embedded in sentences and presented in RSVP for immediate verbatim recall (Kanwisher, 1987). When subjects viewed sentences like "When she spilled the ink there was ink all over", the most common response was something like "When she spilled the ink there was all over". In other words, repetition blindness was strong enough to force subjects to selectively omit the second occurrence of the repeated word, sacrificing the meaning and grammaticality of the sentence.

Repetition blindness can not, however, be due to the first occurrence of a repeated word interfering with *recognition* of the second occurrence. When subjects only have to report the last word in an RSVP list performance is helped, not hindered, by an earlier occurrence of the target word in the same list (Kanwisher, 1987). Thus, repetition blindness cannot be a problem in recognizing both occurrences of a repeated word. Rather, what is difficult at rapid presentation rates is individuating the two occurrences as distinct events, or tokens. (Presumably in the name-the-last-word experiment, it was the last word, not the first, which was individuated, or "tokenized", whereas in sentence recall it is typically the first occurrence which is tokenized at the expense of the second.) Thus, RB is a case in which the second occurrence of the repeated word is recognized as a type but not individuated as a new token.

Repetition blindness can thus be taken as evidence of a dissociation between type recognition and token individuation in visual information processing. If this dissociation is a general property of human vision, rather than a peculiarity of the way rapid word lists are processed, then we might expect to also find repetition blindness in very different kinds of visual stimuli. First, the effect ought to happen not only for complex visual types like words, but also for simple visual features like colors and shapes. Second, RB ought to happen for items presented simultaneously in spatial arrays, as well as items presented sequentially in temporal lists. Thus, the present experiments looked for repetition blindness in simultaneously-presented arrays of colors and letters.[1]

EXPERIMENT: REPETITION BLINDNESS FOR VISUAL FEATURES IN SPATIAL ARRAYS

Subjects viewed briefly-presented arrays of four letters (Exp A) or four color patches (Exp B), chosen from a set of six. The subjects' task was simply to report the two array items indicated by two cue boxes surrounding their locations. Two items were cued rather than one because RB is an inability to individuate two tokens of the same type, rather than an inability to simply individuate one of two identical stimulus tokens. The cues either appeared just before the array (the Precue condition), or immediately after a 200-ms mask which followed the array (the Postcue condition). Subjects were told to report the two items in a pre-specified order (top before bottom; if both were on the same row, then left before right).

The key question was whether, after correctly reporting the first item, subjects would be less likely to report the second item correctly if it was the same (the Repeated condition) than if it was different (the Unrepeated condition). To diminish the possibility of a response bias favoring report of Unrepeated items, the instructions strongly emphasized the existence of trials containing two identical targets. In order to avoid floor or ceiling effects, a staircase procedure was used to adjust the stimulus duration periodically to keep performance at about 50% correct on Unrepeated trials. ("Correct" means both items were reported correctly in the correct order.) Separate staircases for Precue and Postcue conditions were used. In each case, the adjustments were made on the basis of performance on Unrepeated trials only, but Repeated and Unrepeated trial durations were yoked together.

Method

Subjects. Twenty-one UC Berkeley students participated, 16 in the letter version (A) and 16 in the color version (B). Five subjects were excluded because their performance on Unrepeated trials averaged 15% or less correct in either the letter or the color experiment, indicating that they were not far above chance even for the longest allowable stimulus duration. (Even for these subjects, Repeated scores averaged lower than Unrepeated scores.)

---

1 This experiment is reported in more detail in Kanwisher (1990).

Materials, Design, and Procedure. In experiment A, the array items were capital letters drawn from the set E,X,T,O,S, and W. These letters were selected to be as likely as possible to differ in terms of primitive shape features. In Experiment B, the array items were small color patches (composed of two adjacent # signs) drawn from the set red, purple, green, white, yellow, and blue. Experiments A and B were isomorphic (that is, A was translated into B by converting each particular letter into a patch of a particular color). In each experiment, the 96 test trials were broken down into 24 trials in each of the four conditions created by crossing Repeatedness by Cue (Pre versus Post). There were two versions each of Experiments A and B. The design counterbalanced for any effects of particular letters or colors, target locations, or any interaction of these. In addition to the test items, there were 48 filler trials which included repeated colors or letters which were not both probed as target items.

The experiment was carried out on an AST AT computer with a NEC Mutisync II screen, using Psychology Software Tools' MEL experimental software (Schneider, 1988). Each trial began when the subject hit the return key on the computer keyboard. A fixation point appeared for 750 ms in the center of an outline square defining the border of the array. Then in the Precue condition the two small cue boxes appeared inside the outline square for 150 ms, surrounding the location where two of the stimulus items would next appear. Next, the stimulus array composed of four letters or color patches flashed on briefly (one in the center of each quadrant of the large square). The stimulus array was displayed for a variable interval determined by the staircase manipulation. Finally, a mask composed of four rectangular white point arrays (Experiment A) or four rectangular color Mondrians (Experiment B) flashed on for 200 ms covering the locations of the four array items. The Postcue condition was identical except that the cues appeared after the mask, not before the stimulus array.

Subjects were instructed to fixate on the point, look at the array and the cue boxes, and report the two items appearing in the location surrounded by the cue boxes. The subject typed the response into the computer keyboard. This was either the two letters they thought they saw (Exp. A) or the first letters of the names of the two colors they thought they saw (Exp. B). If they had no idea what color or letter was presented in either or both positions, they typed corresponding question marks. Subjects were told three times during the instructions that the experiment contained trials with repeated letters (or colors), and that if they thought both target items were the same they should type the corresponding letter in twice. Before the experimental session, subjects learned the stimulus set by going through 8 trials with feedback which were just like the experimental trials (including both Repeated and Unrepeated trials), except that the stimulus array was displayed for a full second. This served to train them on the color names (and letter set) and to make sure they understood how to respond correctly to Repeated trials. Then they did 24 faster practice trials (without feedback) before the experiment began. Stimulus durations were adjusted periodically throughout the practice test and experimental trials, to keep Unrepeated performance at about 50% correct for both Pre and Post Unrepeated conditions (there were separate staircases for Pre and Post trials).

Results

The results are shown in Table 1. Individual target items were only scored as correct if they were reported in the correct location (indicated by report order). The data were then scored in terms of the conditional probability of getting the second item correct given that the first item was reported correctly. (Performance on the first item alone was fairly constant across conditions--averaging 77% for the letter experiment and 68% for the color experiment--so this technique does not differ much from simply reporting the percent of trials in which subjects got both target items correct.) A

correction was used to discount each subject's raw Unrepeated score by the expected number of correct Unrepeated responses due to guessing, based on an analysis of that subject's errors.[2]

	Precues			Postcues		
	Rep.	Unrep.	Duration	Rep.	Unrep.	Duration
Letters (Exp. A)	.41	.60	62 ms	.44	.48	120 ms
Colors (Exp. B)	.24	.49	88 ms	.45	.46	129 ms

Figure 1. The probability of getting the second item correct, given correct report of the first, as a function of Cue (pre vs. post) and whether the two items are the same (Rep.) or different (Unrep.). The Duration column gives the average stimulus duration for that condition. Unrepeated scores have been corrected (downward) for guessing, as described in footnote 2.

Analysis of variance by subjects on the (corrected) probability of getting the second item correct given correct report of the first showed a significant main effect of Repetition, $F(1,15) = 10.0$, $p < .01$, and significant interactions of Cue x Repetition, $F(1,15) = 16.4$, $p < .01$, and Cue x Color/Letter, $F(1,15) = 8.3$, $p < .05$. However, there was no significant interaction of Repetition x Color/Letter, $F = 0.1$. No other main effects or interactions reached significance.

Discussion

These data show substantial repetition blindness for both colors and letters, but only in the Precue condition. That is, in the Precue condition, given correct report of the first target item, subjects were significantly less likely to get the second item correct when it was the same as the first (e.g., "XX" or "red red"), compared to when it was different (e.g., "TX" or "blue red"). Thus, repetition blindness--and the type-token problem it exemplifies--generalize to spatial displays of simple visual stimuli.

The fact that RB seems to occur primarily in the Precue condition is a bit mysterious. At first glance, one might have predicted the opposite--i.e. that directing attention to the target locations ahead of time (as in the Precue condition) might have diminished RB. However this didn't happen; RB is evidently robust even when subjects know the target locations before the stimulus appears. As for the lack of RB in the Postcue condition, the most straightforward account would be that the stimulus durations necessary to obtain equivalent performance were longer. Alternatively, Precues might somehow pre-empt grouping processes (available in the Postcue condition) which organize the display into "chunks" according to similarity, making repetitions evident by their emergent orientation or other features. More research is needed to explain the sensitivity of RB to cueing strategy.

2 Corrections were made for Unrepeated Pre and Unrepeated Post conditions separately. For each, the number of incorrect responses in which the subject got the first item correct, but reported a different item from the array in the place of the second item was tallied. This count (N) includes 2/3 of all trials in which the subject guessed the second item from a pool of recognized-but-not-located items, 1/3 of which would be expected to generate correct responses. Thus each raw Unrepeated score, i.e. the number in which both the first and second target were correctly reported in the correct order, was discounted by N/2 before dividing by the total number of trials in which the first item was correctly reported. This correction also takes into account second-target outright guesses of items that were neither recognized nor located, since the number of these which should be discounted (i.e. the number of responses in which the first target is correct, but the second is an item not anywhere in the array) happens to equal the number of these responses which get spuriously included in the location-guess (N) count. Conservatively, no correction was made for the Repeated condition data.

CONCLUSIONS

The available evidence suggests that both repetition blindness and illusory conjunctions reflect general limitations of visual information processing. Both phenomena occur for a broad range of visual stimuli - from simple visual features like colors to complex visual entities like words abstracted across case. Both occur for stimulus items distributed across either space or time, and both can be described (Kahneman and Treisman, 1984; Kanwisher, 1987, 1990) as errors in binding visual types (colors, letters, words) to visual tokens (objects or events). That is, RB can be thought of as an error binding one type to two tokens, whereas illusory conjunctions can be thought of as errors binding two different types to one token. Further, Kanwisher (1990) has argued that the tokens necessary to detect visual repetitions are the same mental entities as the tokens necessary to conjoin visual features. Yet, despite all these similarities, the present experiment demonstrates an important asymmetry in the two phenomena.

While illusory conjunctions can be explained in terms of an overall lack of location information, repetition blindness cannot. In the current experiment, items were only counted as correct when they were reported in the correct location, so both Repeated and Unrepeated trials required subjects to bind item identities to their locations. Thus, a general binding problem or an overall lack of positional information (as implicated by Mozer, 1989 and Keren & Boer, 1985) can not explain the observed difference in performance in the Repeated and Unrepeated conditions. (If anything, a general binding problem would favor the Repeated condition, because switches between the two target locations would not be detected, whereas they would be counted as incorrect in the Unrepeated condition.) Instead, there seems to be a particular difficulty in binding one type to two different tokens, above and beyond the difficulty of binding two different types to two different tokens (or, it would seem, in binding two different types to one token, as in feature conjunction). The early experiments on RSVP word lists demonstrate the analogous situation for temporal tokens: temporal RB was robust even though the serial order of unrepeated words was reported fairly accurately.

Thus, human visual perception seems to suffer from two distinct problems: a binding problem and a type-token problem.3 Both problems are typical of connectionist networks but not of symbol-manipulating systems, which suggests that connectionist networks may be better able to model some stages of human visual information processing. If so, then it is perhaps noteworthy that repetition blindness does not generalize indefinitely. For example, there is no RB at the level of meaning: when two synonyms are embedded in a sentence and presented in RSVP for immediate recall, both are reported with ease (Kanwisher and Potter, 1990).[4] This might suggest an upper bound for the utility of connectionist models of visual processing, somewhere after lexical entries are activated, yet before meanings are retrieved.

ACKNOWLEDGEMENTS

The research reported here was supported by NIMH grant MH45245-02 to the author. I thank Macy Ng for research assistance, and Molly Potter, John Rubin, Michael Jordan, and Michael Mozer for heroic last-minute commentary. Correspondence should be sent to Nancy Kanwisher, Department of Psychology, University of California, Berkeley, Ca 94720.

---

3 An alternative perspective suggested to me by M. Mozer (personal communication, March 1990) is also consistent with the present data and explains the type-token problem as a by-product of the sequential "readout" system which does binding: if an item's identity is suppressed after it is read out, the difficulty with repetitions will result.

4 Also, when sentences containing repeated words are presented auditorily in compressed speech, no "repetition deafness" is found (Kanwisher and Potter, 1989).

Bjork, E. L., & Murray, J. T. (1977). On the nature of the input channels in visual processing. Psychological Review, 84, 472-484.

Fodor, J.A. & Pylyshyn, Z. W. (1988). Connectionism & cognitive architecture: A critical analysis. Cognition, 28, 3-71.

Feldman, J. A. & Ballard, D. H. (1982). Connectionist models and their properties. Cognitive Science, 6, 205-254.

Hinton, G. E. & Lang, K. J. (1985). Shape recognition and illusory conjunctions. Proceedings of the International Joint Conference on Artificial Intelligence, Vancouver, B.C., Canada.

Hinton, G. E., McClelland, J. L., & Rumelhart, D.E. (1986). Distributed Representations. In Parallel distributed processing: Explorations in the microstructure of cognition. Vol 1:Foundations, ed. D. E. Rumelhart, J. L. McClelland, & the PDP Research Group. MIT Press/ Bradford Books.

Intraub, H (1989). Illusory conjunctions of forms, objects, and scenes during rapid serial visual presentation. Journal of Experimental Psychology: Learning, Memory, and Cognition, 15, 98-109.

Kanwisher, N. (1986). Repetition blindness: Type recognition without token individuation. Unpublished doctoral dissertation, Massachusetts Institute of technology, Cambridge.

Kanwisher, N. (1987). Repetition blindness: Type recognition without token individuation. Cognition, 27, 117-143.

Kanwisher, N. (1990). Repetition blindness and illusory conjunctions: Errors in linking visual types with visual tokens. Manuscript submitted for publication.

Kanwisher, N. & Potter, M. (1989). Repetition blindness: The effects of stimulus modality and spatial displacement. Memory and Cognition, 17, 117-124.

Kanwisher, N. & Potter, M. (1990). Repetition blindness: Levels of processing. Journal of Experimental Psychology: Human Perception and Performance, 16, 30-47.

Kahneman, D & Treisman, A. (1984). Changing views of attention and automaticity. In R. Parasuraman & D. R. Davies (Eds.), Varieties of Attention. New York: Academic Press.

Keele, S.W., Cohen, A., Ivry, R., Liotto, M., and Yee, P. (1988). Tests of a temporal theory of attentional binding. Journal of Experimental Psychology: Human Perception and Performance, 14, 444-452.

Keren, G. & Boer, L. C. (1985). Necessary and sufficient conditions for repeated-letter inferiority: The role of positional uncertainty. In M. I. Posner & O. S. M. Marin (Eds.), Attention and Performance XI, 601-612, Hillsdale, NJ: Erlbaum.

Koch, C & and Ullman, S. (1985). Shifts in selective visual attention: towards the underlying neural circuitry. Human Neurobiology, 4, 219-227.

Lawrence, D. H. (1971). Two studies of visual search for word targets with controlled rates of presentation. Perception and Psychophysics, 10, 85-9.

McLean, J. P., Broadbent, D. E., & Broadbent, M. H. P. (1982). Combining attributes in rapid serial visual presentation tasks. Quarterly Journal of Experimental Psychology, 35A, 171-186.

McClelland, J. L. (1986). The programmable blackboard model of reading. In Parallel distributed processing: Explorations in the microstructure of cognition, Vol 2: Psychological and Biological Models, ed. J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/ Bradford Books.

Mozer, M. (1987). Early parallel processing in reading: A connectionist approach. In M. Coltheart, Attention and Performance XII, Hillsdale, NJ: Lawrence Erlbaum Associates.

Mozer, M. (1988). A connectionist model of selective attention in visual perception. Proceedings of the Tenth Annual Conference of the Cognitive Science Society (pp. 195-201).

Mozer, M. (1989). Types and tokens in visual letter perception. Journal of Experimental Psychology: Human Perception and Performance, 15, 287-303.

Norman, D. A. (1986). Reflections on cognition and parallel distributed processing. In Parallel distributed processing: Explorations in the microstructure of cognition, Vol 2: Psychological and Biological Models, ed. J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/ Bradford Books.

Pinker, S. & Prince, A (1988). On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. Cognition, 28, 73-193.

Pylyshyn, Z. W. (1989). The role of location indexes in spatial perception: A sketch of the FINST spatial-index model. Cognition, 32, 65-97.

Rumelhart, D. E. & Norman, D. A. (1982). Simulating a skilled typist: A study of skilled cognitive-motor performance. Cognitive Science, 6, 1-36.

Santee, J. L. & Egeth, H. E. (1980). Interference in letter identification: A test of feature-specific inhibition. Perception and Psychophysics, 27, 321-330.

Schneider, W. (1988). Micro experimental laboratory: An integrated system for IBM PC compatibles. Behavior Research Methods, Instruments, and Computers, 20, 206-217.

Smolensky, P. (1987). A method for connectionist variable binding. Technical Report CU-CS-356-87, Department of Computer Science, University of Colorado at Boulder.

Strong, G. W. & Whitehead, B. A. (1989). A solution to the tag-assignment problem for neural networks. Behavioral and Brain Sciences, 12, 381-433.

Touretzky, D. S. & Hinton, G. E. (1986). A distributed connectionist production system. Technical Report CMU-CS-86-172, Computer Science Department, Carnegie Mellon University.

Treisman, A. & Schmidt, H. (1982). Illusory conjunctions and the perception of objects. Cognitive Psychology, 14: 107-41.

Treisman, A. & Gelade, G. (1980). A feature integration theory of attention. Cognitive Psychology, 12, 97-136.

Dynamic Binding: A Basis for the Representation of Shape by Neural Networks[1]

John E. Hummel and Irving Biederman

Department of Psychology, University of Minnesota, Minneapolis, MN 55455

Abstract

A neural network model for object recognition based on Biederman's (1987) theory of Recognition by Components (RBC) is described. RBC assumes that objects are recognized as configurations of simple volumetric primitives called *geons*. The model takes a representation of the edges in an object as input and, as output, activates an invariant, entry-level representation of the object that specifies the object's component geons and their interrelations. Local configurations of image edges first activate cells representing local viewpoint-invariant properties (*VIPs*), such as vertices and 2-D axes of parallelism and symmetry. Once activated, VIPs are bound into sets through temporal synchrony in the firing patterns of cells representing the VIPs and image edges belonging to a common geon. The synchrony is established by a mechanism which operates only between pairs of a) collinear, b) parallel, and c) coterminating edge and VIP cells. This design for perceptual organization through temporal synchrony is a major contribution of the model. A geon's bound VIPs activate independent representations of the geon's major axis and cross section, location in the visual field, aspect ratio, size, and orientation in 3-space. The relations among the geons in an image are then computed from the representations of the geons' locations, scales and orientations. The independent specification of geon properties and interrelations uses representational resources efficiently and yields a representation that is completely invariant with translation and scale and largely invariant with viewpoint. In the final layers of the model, this representation is used to activate cells that, through self-organization, learn to respond to individual objects

Introduction

Within the limits of visual resolution, and excluding so-called "accidental" viewpoints (i.e., singularities of viewing angle that project misleading images on the retina, such as viewing a cylinder from an angle that makes it appear to be a rectangle), an object's image may be projected on the retina in any location, in any size, and from any viewing angle, and the object will still be readily recognized. Biederman's (1987) theory of Recognition by Components (RBC) explains these fundamental phenomena of object recognition by positing that objects are represented as structured configurations of viewpoint-invariant volumetric primitives called *geons*. This paper introduces a neurally plausible model of object recognition based upon RBC. Although the model described herein is a complete model of recognition, this paper describes only those parts explicitly concerned with the deriviation of an object's structural description in terms of geons and their relations.

To derive a viewpoint-invariant representation of the geons and relations in an image of an object, a neural network (NN) must solve three related problems: (1) For any image

---

[1]This research was supported by AFOSR Research Grant 88-0231 to I.B. and an NSF Graduate Fellowship to J.E.H. Correspondences should be addressed to J.E.H. (Bitnet: EQZ6628@UMNACVX) or I.B. (Bitnet: PSYIRV@UMNACVX)

containing more than one geon, it must determine which image features belong with which geons; (2) it must recognize the geons and represent them in a manner that, while invariant with location and viewpoint, expresses the location and orientation of each geon; and (3), it must derive the relations among the different geons in the image and bind those relations to the geons to which they apply. These tasks are all manifestations of the *Binding Problem*, a problem that has not been adequately handled by artificial neural networks. We describe a solution to binding which allows the present model to solve each of these problems.

The Binding Problem The term *binding* refers to the represention of feature conjunctions. For example, how can a NN represent an image edge that is at a particular location in the image *and* at a particular orientation *and* with a particular curvature, etc.? The predominant approach is to allocate a cell (or specific pattern of activity over a set of cells) to respond to edges with the desired combination of properties. Likewise, other cells or patterns would be allocated to respond to all other combinations. We will use the term *enumerated* to refer to representations of this type because feature conjunctions are represented by enumerating all possible combinations and allocating separate cells for each.

Despite its popularity in NNs, enumeration suffers critical shortcomings as a general solution to binding. Its most serious difficulty is that the cells representing a given feature conjunction must be dedicated *prior* to the occurrence of that conjunction in the system's input. In addition to inefficiency of representation (most cells are unused most of the time), this requirement precludes *dynamic* binding. *Dynamic binding* refers to conjoining stimulus properties that are represented in *different* cells or even different parts of the brain. The problem of dynamic binding is typically illustrated in the context of conjoining an object's color and shape, and its solution is usually described in terms of an attentional mechanism that operates by somehow "gluing" together different properties that are linked to a common point in some sort of "location map" (e.g., Kahneman & Treisman, 1984; Treisman & Gelade, 1980).

But the problem of dynamic binding has implications far beyond conjoining color and shape by reference to common location. First, any image projected on the retina will exist over a range of locations, so even assembling the various features defining a shape (Problem 1 above) entails binding features occurring at *different* locations. In this context, binding is referred to as *image parsing* or *perceptual grouping* (although whether the binding is performed dynamically or by pre-dedicated cells is rarely addressed explicitly). Representing geons in a manner invariant with location and viewpoint while still expressing these properties (Problem 2) also requires a mechanism for dynamic binding since, to be invariant with location, the representation of a geon must be independent of the represention of its location. Therefore, conjoining these separate representations to express the location of a particular geon requires dynamic binding. The same logic applies to binding geons and relations (Problem 3). Unless a different cell is to be posited for each possible geon in each possible relation with every other geon, binding geons and relations entails dynamically binding features represented in separate cells. Thus, the dynamic binding problem underlies each of the above difficulties posed by a NN approach to viewpoint-invariant recognition.

Binding Through Synchrony In the present model, independent features are dynamically bound by establishing synchronous firing in the cells representing those features. Although synchrony as a basis for binding was first described by von der Malsburg (1981,1987) and later by others (e.g., Crick, 1984), this article presents an original proposal for establishing synchrony among the basic features of complex shapes. Specifically, we posit the existence of *Fast Enabling Links* (*FELs*) that induce

synchronous firing in active units sharing them. In the model's first two layers, visual features represented by cells sharing FELs are grouped into coherent shapes. The form of the grouping, fundamental to the model's capacity for representing shape, is determined by the specific set of FELs. The resulting synchrony is then used in higher layers both to bind the independent properties of geons, and to bind relations to the geons they describe.

The Model

Overview The complete model is a 7 layer connectionist network that takes as input a representation of a line drawing of an object and, as output, activates a cell representing the entry-level category of the object. An overview of its architecture is shown in Figure 1. The model's first layer (L1) is composed of a mosaic of cell clusters distributed over the model's visual field. The cells within an L1 cluster respond to image edges in terms of their orientation, curvature, and whether they terminate within the cluster's receptive field. The model's second layer (L2) is also composed of a mosiac of cell clusters. These cells respond to configurations of edges that define vertices, 2-D axes of parallelism and symmetry, and oriented, elongated blobs at particular locations in the visual field. Cells in L1 and L2 group themselves into sets (or assemblies) describing geons by establishing temporal synchrony among their spikes of output. Cells tend to fire in synchrony if they represent features of the same geon and out of synchrony if they represent features of different geons.

Cells in L3 respond to properties of complete geons. These cells take their input directly from L2, but because of the binding achieved in L1 and L2, each of the geon properties is represented independently of every other property. For example, the shape of a geon's major axis (straight or curved) is represented in one vector of cells and the geon's location is represented in another vector. Consequently, the representation of the geon's axis does not change when the geon is moved in the visual field. The fourth and fifth layers compute the relations among the geons represented in L3. These computations are performed on the basis of the geons' coarsely coded metric properties (i.e., location in the visual field, scale and orientation in 3-space). The relations among the geons are bound to the geons they describe by the same synchrony of firing that binds image features together for geon recognition. The output of layers 3 and 5 together describe an object in terms of its geons and their relations. This representation is invariant with scale and translation, and largely invariant with viewpoint (orientation in the visual plane and in depth).

The representations and processes employed in layers 1 to 5 constitute the major contribution of this effort, but the model has two additional layers that use the output of layers 3 and 5 as a basis for invariant recognition. These layers integrate the outputs of L3 and L5 over time and self-organize to recognize complete objects. This paper will emphasize the processes employed in L1 and L2 for image parsing and the processes employed in L4 for computing relations.

Image Parsing

Image parsing is among the first problems to confront a geon-based model of visual recognition because the VIPs in an image must be grouped before the geons they define can be identified. For example, correctly parsing the image in Figure 2a entails grouping vertices V1,V2 with segments S1,S2 as features of one geon, and V3 with S3 as features of the other. This model performs parsing by establishing temporal synchrony among the cells in L1 and L2 representing features of a common geon. The synchrony is established through local interactions among edge- and VIP-sensitive cells.

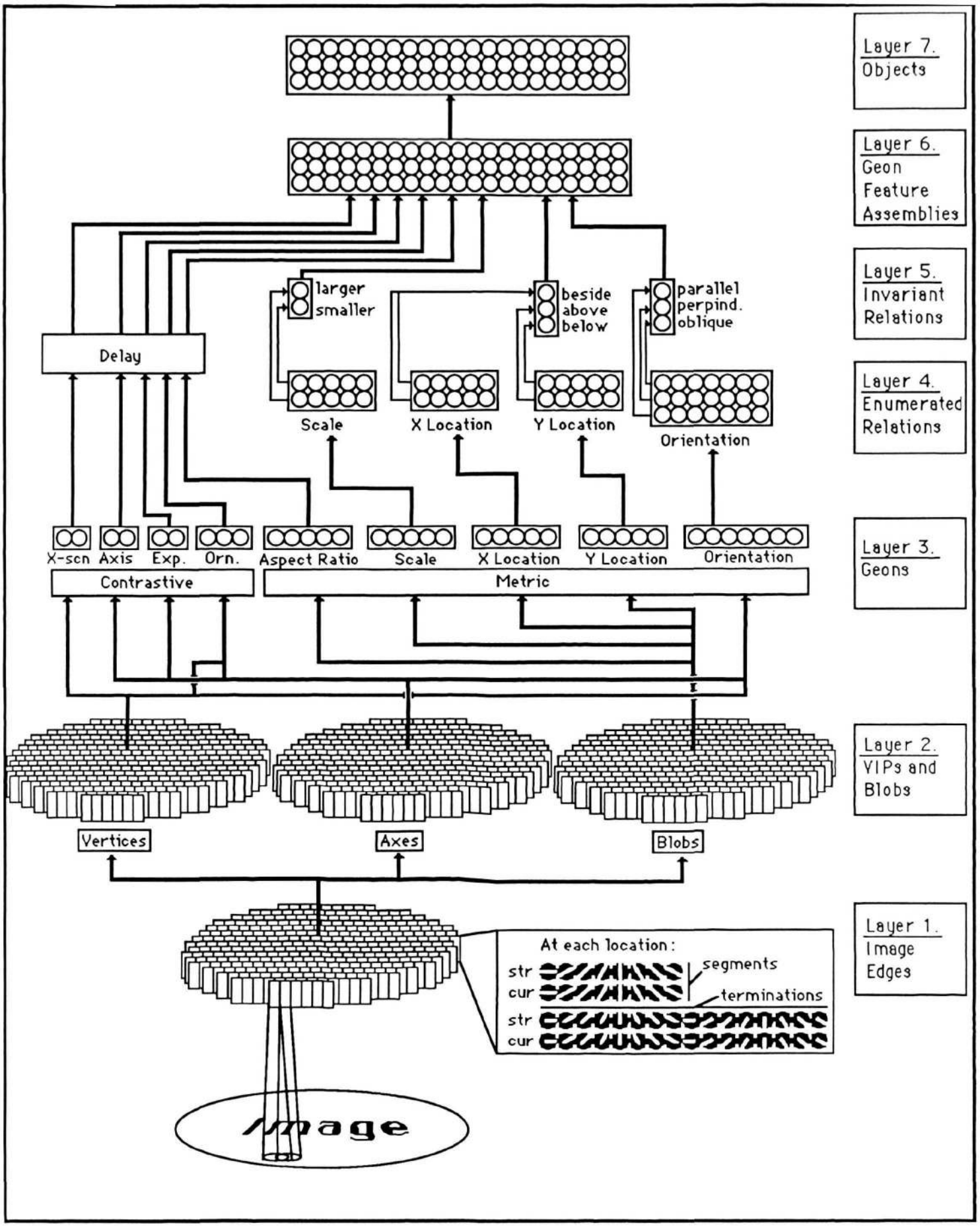


Figure 1. Overall Architecture

Cells in L1 share two types of connections with other cells in the network: typical *activation* connections (or simply *connections*), which spread excitation and inhibition from one cell to another, and *fast enabling links* (*FELs*), which are assumed to operate approximately an order of magnitude faster than the duration of a "time slice" (the temporal period within which cells sum their inputs), and which propagate only binary *enabling* signals between cells. FELs induce synchronous firing in pairs of active cells as follows: Each $cell_i$ in L1 has an output refractory R_i which prevents it from generating a continual train of output spikes. If $cell_i$ is active, it will generate a spike of output only when R_i goes below the refractory threshold (0). R_i is assumed to decay linearly. When $R_i \leq 0$, it is reset to its maximum, and $cell_i$ fires (it generates a spike of output and sends a signal out along all its FELs). Because of the speed with which FELs propagate, an enabling signal will tend to arrive at its destination within the same time slice it was generated. When it arrives at an active $cell_j$, its effect is to set R_j immediately to zero, causing $cell_j$ to go through the same sequence of events as $cell_i$ (i.e., reset its refractory, and generate an output and an enabling signal). If the enabling signal arrives at an inactive cell, nothing happens.

Because active cells sharing FELs tend to fire in synchrony, they organize themselves into groups defined by their temporal firing patterns. As shown in figure 2b, FELs are posited between five types of cell pairs in L1 and L2, reflecting four general constraints on the formation of edge-based images:

1) Image edges usually extend beyond the receptive field of a single edge-sensitive neuron, so a given image edge will tend to excite collinear, adjacent edge-sensitive cells. Such cells are synchronized by FELs *a* and *b* in Figure 2b.
2) The receptive fields of adjacent edge-sensitive cells overlap, so a given image edge will tend to excite parallel, adjacent edge-sensitive cells. These cells are synchronized by FEL *c*.
3) Edges that coterminate in an intra-geon vertex (a fork, arrow, L or tangent Y) tend to correspond to edges of a common geon. These features are grouped by FELs between termination and vertex cells in corresponding locations of L1 and L2 (FEL *d*).
4) An edge occluded by a surface will appear as collinear segments that, excluding accidental alignments, do *not* coterminate with other edges at the points of occlusion. Complimentary, collinear termination cells in distant locations are synchronized by FEL *e* in Figure 2b.

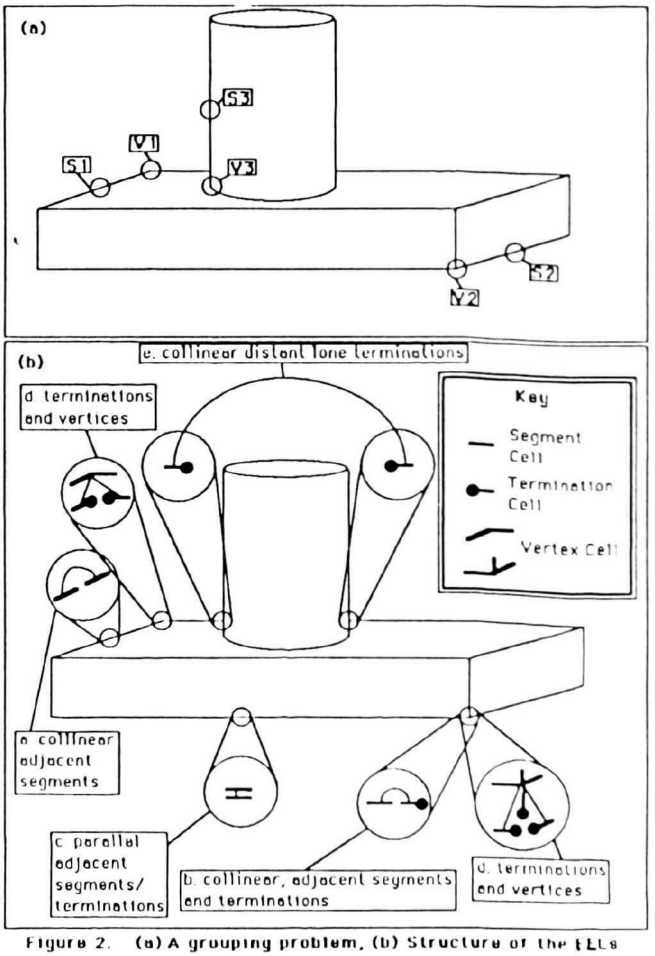


Figure 2. (a) A grouping problem. (b) Structure of the FELs

FELs *a*, *b* and *c* cause all the cells representing a continuous edge to fire synchronously. Termination-to-vertex FELs (d) group vertices with the edges to which they are attached. Since FELs are bidirectional, all the edges coterminating at a given vertex will also be grouped together (an enabling signal will enter a vertex from one termination and be passed, via that vertex, to the other terminations at that location). The distant termination-to-termination FELs (e) operate through vertex cells in L2 which respond specifically to *lone terminations* (edges that do not terminate with other edges, such as the stem of a T vertex). FELs between collinear lone terminations allow the visible pieces of occluded edges to group with one another. Because the L2 lone termination cells are inhibited by inconsistent terminations in L1, this type of "gap jumping" cannot occur when more than one edge terminates at a point. This restriction prevents edges belonging to different geons from being grouped just because they happen to be collinear. Note also that this set of FELs implicitly excludes grouping edges that form T vertices. T vertices are formed at the junction of separate geons, the "top" belonging to one geon, and the "stem" to another, and therefore constitute image features whose constituent parts should *not* be grouped.

Applied iteratively, locally grouping edges and vertices causes all features belonging to a common geon to be grouped, and since blobs and axes receive their inputs from edge cells, the blobs and axes belonging to a given geon will fire on the same time slices as the geon's edges and vertices. In this manner, the local computations performed by the FELs parse an image into its constituent geons. Unlike a top-down or knowledge-driven mechanism, these computations require no information about what volumes are in the image and where they are located. This is an important advantage, because a parsing mechanism that required such information would effectively require that the image already be parsed.

Representing Geons The cells of the model's third layer represent the properties of geons (shown in Figure 1) which have the following characteristics: (1) Geon properties are divided into two classes: *contrastive* properties (such as straight axis vs. curved axis) and *metric* properties (such as location in the image). The former are used directly for recognition while the latter are used to compute the relations among the various geons in an image. (2) Geon properties are activated by the VIPs activated in layer 2. For example, all L2 cells representing Tangent Y vertices activate the cell which responds to curved cross section geons. (3) Each L3 cell responds independently to a particular value on a particular dimension over which geons can vary. For example, the L3 cell that responds to the value *curved* on the dimension *shape-of-major axis* will fire in response to any geon with a curved axis, such as a large curved brick in the upper left of the visual field or a small curved cone in the lower right. Thus, *each geon property is represented in a manner that is invariant with every other geon property.* This invariance, made possible by the binding achieved in L1 and L2, is a crucial aspect of the model's design.

Deriving Relations Among Geons Of the properties derived in L3, only the contrastive properties are used directly for geon classification. The metric properties (size, location, and orientation) are used to determine the relations among the geons in the image, such as relative size, relative location, and relative orientation (Figure 1).

Determination of inter-geon relations is performed in two steps. In L4, relations are computed separately for each value of each dimension. The relation *below* will be used to illustrate how the L4 cells operate, but the logic generalizes to all relations. Consider Figure 3. Associated with every position in Y (Y_p), there is an L3 cell which becomes active when that position is occupied by a geon ($L3_{y=p}$), and there are two L4 cells: one that becomes active when Y_p is below another occupied position ($L4_{\text{below at } y=p}$), and

one that becomes active when Y_p is above another occupied position ($L4_{above\ at\ y=p}$). In L5, there is only one cell for each relation, each of which receives excitation from every corresponding L4 cell. For example, $L5_{below}$ receives excitation from $L4_{below}$ at $y=1$, $L4_{below}$ at $y=2$, etc. Each L4 cell receives two types of input: an excitatory input and an enabling signal (through an FEL). $L4_{below}$ at $y=1$ receives an enabling singal from $L3_{y=1}$ and excitatory input from $L3_{y=2} \ldots L3_{y=5}$. To determine its activation, each L4 cell sums its excitatory inputs over time. Suppose there is a geon at Y_1 ($geon_1$) and another at Y_3 ($geon_3$). $L3_{y=1}$ will fire on the same time slices as the other properties of $geon_1$, and $L3_{y=3}$ will fire in synchrony with the other properties of $geon_3$. $L4_{below}$ at $y=1$ receives an excitatory input from $L3_{y=3}$ because Y_1 is below Y_3. Therefore, when $L3_{y=3}$ fires, $L4_{below}$ at $y=1$ will receive an excitatory input, and its activation will go above zero. When $L3_{y=1}$ fires, it sends an enabling signal to $L4_{below}$ at $y=1$, causing it to fire. Note that $geon_1$ is positioned below $geon_3$. *Below* is therefore a property that describes $geon_1$, and $L4_{below}$ at $y=1$ is now firing in synchrony with $geon_1$'s other properties. Since $L4_{below}$ at $y=1$ sends an excitatory signal to $L5_{below}$, $L5_{below}$ will also fire in synchrony with $geon_1$'s other properties.

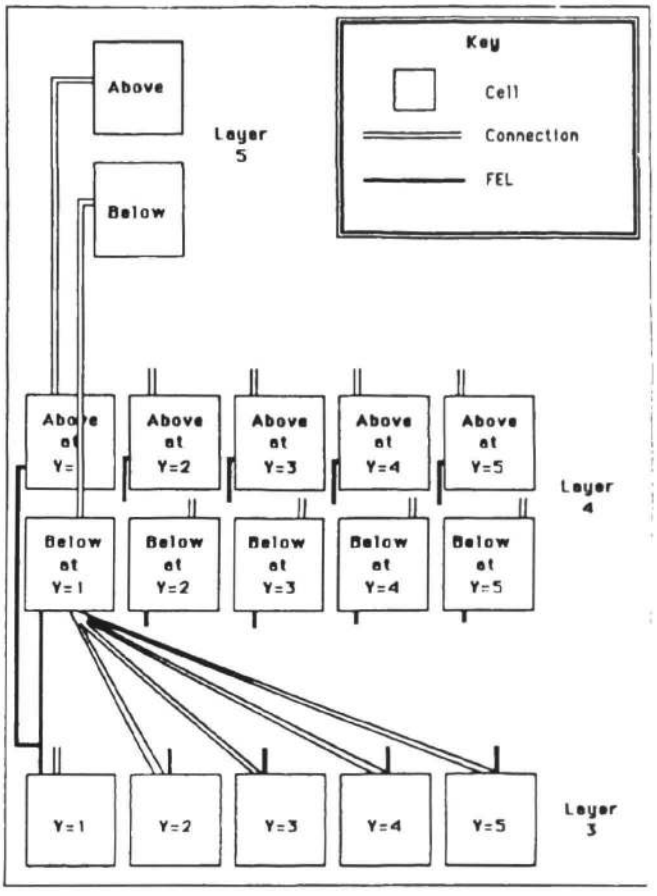


Figure 3. Computing Relations

Recognizing Objects The model's sixth layer receives inputs from L3 cells describing geons, and from L5 cells describing inter-geon relations. L6 and L7 perform two functions: they use the temporal synchrony of inputs within an assembly to ensure that the geons in a object are in the appropriate configuration to define that object, and they combine information from different time slices into an interpretation of a single object.

Preliminary Results and Discussion

Simulations with the model described here have shown that the model is capable of parsing line drawings of simple (even unfamiliar) objects and deriving descriptions of their geons and relations that are completely invariant with scale and translation, and largely invariant with viewpoint. As a consequence, it demonstrates complete translation and scale invariance in recognizing each of the objects with which it is familiar (the objects on which it was allowed to self-organize), and demonstrates rotation invariance resembling that of the human in experimental situations with nonsense objects. That is, it tolerates rotations in depth better than rotations in the visual plane, and its performance on rotations in the plane degrades as a function of the degree of rotation.

References

Biederman, I. (1987). Recognition by components: A theory of human image understanding. *Psychological Review*, 94, 2, 115-147.

Crick, F. H. C. (1984). The function of the thalamic reticular spotlight: The searchlight hypothesis. *Proceedings of the National academy of Sciences*, USA 81, 4586-4590.

Kahneman, D. & Treisman, A. (1984). Changing views of attention and automaticity. In R. Parasuraman, R. Davies, & J. Beatty (Eds.) *Varieties of Attention* (pp. 29-62). New York: Academic Press.

Treisman, A. & Gelade, G. (1980). A feature integration theory of attention. *Cognitive Psychology*, 12, 97-136.

von der Malsburg, C. (1981). The correlation theory of brain function. Internal Report 81-2. Department of Neurobiology, Max-Plank-Institute for Biophysical Chemistry

von der Malsburg, C. (1987). Synaptic plasticity as a basis of brain organization. In J. P. Chaneaux & M. Konishi (Eds.), *The Neural and Molecular Bases of Learning* (pp. 4111-432). New York: Wiley

Caricature Recognition in a Neural Network[1]

James W. Tanaka

Carnegie-Mellon University

Abstract

In a caricature drawing, the artist exaggerates the facial features of a person in proportion to their deviations from the average face. Empirically, it has been shown that caricature drawings are more quickly recognized than veridical drawings (Rhodes, Brennan, & Carey, 1987). Two competing hypotheses have been postulated to account for the *caricature advantage*. The *caricature hypothesis* claims that the caricature drawing finds a more similar match in memory than the veridical drawing because the underlying face representation is stored as an exaggeration. The *distinctive features hypothesis* claims that the caricature drawing produces speeded recognition by graphically emphasizing the distinctive properties that serve to individuate that face from other faces stored in memory. A computational test of the two hypotheses was performed by training a neural network model to recognize individual *face* vectors and then testing the model's ability to recognize both caricaturized and veridical versions of the face vectors. It was found that the model produced a higher level of activation to caricature face vectors than to the non-distorted face vectors. The obtained caricature advantage stems from the model's ability to abstract the distinctive features from a learned set of inputs. Simulation results were therefore interpreted as support for the distinctive features hypothesis.

Introduction

In a caricature drawing, the artist graphically exaggerates those features of a famous individual in proportion to their deviations from the normative or average face.

---

[1]This research was supported by a NIMH NRSA Training Grant #1T32 MH19102. I would like to thank Martha Farah, Steve Keele, Jay McClelland and Kris Taylor for their helpful suggestions and advice. Please address all correspondence to: Jim Tanaka, Department of Psychology, Carnegie-Mellon University, Pittsburgh, PA, 15213. E-mail address: tanaka@psy.cmu.edu.

For example, in political cartoons, George Bush is usually portrayed as having an especially long chin and Richard Nixon is recognizable by the distinctive slope of his nose. The goal of the caricaturist is to exploit the facial characteristics that serve to individuate that person from everyone else. Gibson (1973) noted that "the caricature may be a poor projection of his face but good information about it. The form of the face is distorted, but not the essential features of the face" (p.6). The relative ease with which caricatures are recognized presents a paradox of sorts. Why is it that these *distorted* renderings are as easily recognized as *veridical* drawings? One explanation has been to suggest that normal face recognition involves some type of caricature process. Like the caricature artist, face recognition processes exaggerate the distinctive features of an individual relative to a normative prototype before storing the representation in long term memory. According to the *caricature hypothesis*, faces are represented in memory not as veridical representations, but as caricatures. Hence, caricature drawings serve as better retrieval cues than veridical drawings because they are more similar to the underlying stored representation. Recent work by Rhodes, Brennan and Carey (1987) has provided some support for the caricature position. They found that caricature drawings of familiar individuals were recognized faster than veridical drawings. It has also been shown that subjects judge caricatures as depicting better likenesses of individuals than veridical drawings (Rhodes et al., 1987) and caricatures are just as likely to be falsely recognized as veridical drawings (Mauro & Kubovy, 1988). Thus, results showing a caricature *advantage* suggest that the caricature drawings may find a more similar match to the stored face representation than the veridical drawings.

The *distinctive features hypothesis* also claims that individual faces are encoded in terms of their distinctive deviations from the normative face. However, in contrast to the caricature hypothesis, the distinctive features position maintains that the deviations are not stored as exaggerated representations, but as veridical ones. In support of this view, past studies have shown that attending to distinctive facial characteristics improves face recognition (Winograd, 1981) and highly distinctive, atypical faces are better remembered than less distinctive, typical faces (Barlett, Hurry & Thorley, 1984; Light, Kayra-Stuart & Hollander, 1979). According to the distinctive features approach, the caricature advantage can be explained in terms of the drawing's stimulus properties, and not in terms of a distorted memory representation. Acting as a kind of "super stimulus," the caricature drawing directs the viewer's attention to distinctive properties of the face thereby facilitating quick access to the stored face representation.

Does the caricature advantage depend on the encoding and storage of a distorted caricature representation or can the advantage be shown based on distinctive feature information alone? In the following simulation, a test of the caricature and distinctive feature hypotheses was performed. A neural network model was trained to recognize the feature vectors describing three different *faces*. After learning, veridical

and caricature feature vectors were presented to the model and their output activations measured. Under these learning conditions, the caricature hypothesis would predict the absence of a caricature advantage because the exaggerated caricature representations were not explicitly encoded in memory. The distinctive features hypothesis predicts a caricature advantage provided that the network is able to abstract the distinctive feature information from the set of learned vectors.

Description of Caricature Model

As shown in Figure 1, the Caricature Model is a single layer connectionist network consisting of eight input units and three output units. The eight input units comprised the feature vector taking on the activation values of either 0 or +1. The three output units represented three hypothetical face units (i.e., Joe, Tom, Bob). Their activation was calculated by the linear activation function:

$$face_i = \sum_{j=1}^{8} i_j w_{ij}$$

where $face_i$ is the activation level of an individual face, i_j is activation of the input units, and w_{ij} is the strength of the connection weights between the feature units and the face unit.

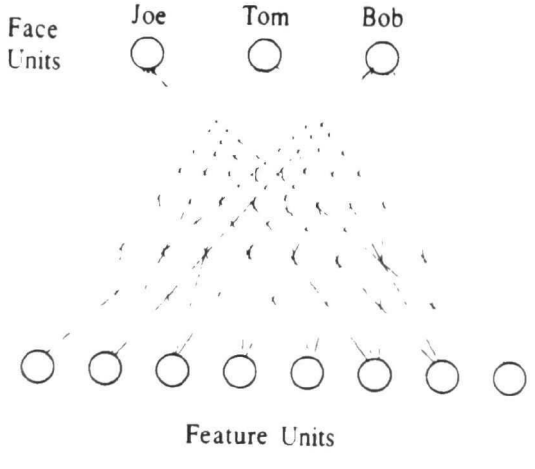


Figure 1. Caricature Model

The network was trained to recognize three prototypical feature vectors, [1,1,1,1,0,0,0,0], [1,1,0,0,1,1,0,0], [1,0,1,0,1,0,1,0], representing the face units of Joe, Tom, and Bob respectively. Instead of presenting the prototypical feature vectors of Joe, Bob and Tom for learning, eight permutations of the prototypical vectors were created by flipping the value of one of the eight input units either to one or zero. For example, a permutation of the first feature in Joe prototype vector would be the feature

vector of [0,1,1,1,0,0,0]. The permuted feature vectors represented the slight perturbations of an individual's face which may arise due to changes in facial expressions or viewing conditions. A total of 24 feature vectors, eight permutations of the three prototypical feature vectors, were presented randomly for learning to the network. Learning was accomplished by adjusting the connection weights between the feature units and the face units according to the standard delta rule (Rumelhart, Hinton & McClelland, 1986):

$$\Delta w_{ij} = \varepsilon (t_i - face_i) \; i_j$$

where Δw_{ij} is the change in connection weights, ε is the learning rate parameter, t_i is the expected output, $face_i$ is the activation of the face unit, i_j is the activation level of the feature unit. Learning of each vector continued until a combined absolute error value of .10 was achieved.

Simulation Results

An initial test was performed to explore whether the model was able to abstract the prototypical patterns from the set of permuted feature vectors. As a test of prototype abstraction, the network's output activations to the 24 learned feature exemplar vectors were compared to the output activations to three prototypical feature vectors. The three prototypical feature vectors produced higher output activations than 23 of the 24 learned feature vectors. Thus, consistent with similar distributed models of memory (McClelland & Rumelhart, 1987), the Caricature Model's was able to demonstrate prototype abstraction by computing a measure of central tendency from a set of given inputs.

An assumption common to the caricature and distinctive features hypotheses is the existence of a normative face representation. That is, both approaches assume that faces are encoded with respect to their deviations from an average representation. As a test for the presence of the normative face representation, the values of the eight feature units were averaged across the three prototypical feature vectors resulting in a "normative" feature vector of [1,.66,.66,.33,.66,.33,.33,0]. When presented to the network, the normative feature vector produced levels of activation that were roughly equivalent in the three face units. Specifically, Joe captured 44% of the available activation, Bob captured 25% of the available activation and Tom captured 30% of the available activation. Joe's higher level of activation was due to the "recency effects" of the model; that is, a Joe feature vector happened to be the last face learned and therefore, exerted a stronger effect on recognition. Thus, all three face units were partially activated by the normative feature vector with none of face units collecting the

majority of available activation. As a consequence of learning the pattern of activations particular to the faces of Joe, Bob and Tom, the Caricature Model indirectly encoded the pattern of activations that described the average face.

The critical test of the caricature and distinctive feature hypotheses was the model's response to the caricature feature vectors. The caricature hypothesis predicts no advantage for the caricature feature vectors because the distorted vectors were not encoded in memory. The distinctive features hypothesis predicts a possible caricature advantage depending upon the model's ability to identify distinctive features of the learned prototypes. As a test of these two hypotheses, caricature versions of the prototypical feature vectors were produced by the following equation:

$$caricature_j = i_j + \beta\ (i_j - norm_j)$$

where $caricature_j$ is the new caricature feature value, i_j is the activation value of the original feature unit, β is a caricature constant that controls the amount of exaggeration, and $norm_j$ is the value of the normative feature. The caricature equation is similar to the equation used in Brennan's Caricature Generator Program (Brennan, 1984). The caricature equation has the property of emphasizing features in proportion to the deviations from the norm. Features that show large deviations from the average face are weighted more heavily than those with small deviations. Applying the formula to the prototypical features with a β of .25 produced caricature feature vectors of [+1,+1.08, +1.08,+1.16,-.16,-.08,-.08,0], [+1,+1.08,-.16,-.08,+1.08,+1.16, -.08,0], [+1,-.16,+1.08,-.08,+1.08,-.08,+1.16,0] for the Joe, Bob, and Tom prototype vectors respectively.

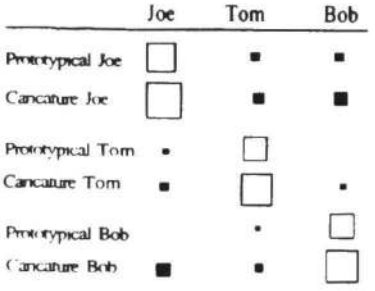


Figure 2. Output activations of the prototypical feature vectors and the caricature feature vectors. Unfilled squares indicate positive activation and filled squares indicate negative activation; the amount of activation is indicated by the area of the square.

Consistent with the prediction of the distinctive feature hypothesis, the caricature feature vectors generated a stronger activation level than the three original prototypical feature vectors (shown in Figure 2). Importantly, not only was the activation of the correct face unit enhanced, but suppression of the incorrect face units was also

increased. Thus, the model demonstrated a caricature advantage without explicitly encoding the caricature representation. What was the source of the caricature advantage? Inspection of the connection strengths between the feature units and the face units revealed that the model assigned a larger weight value to those connections that were the most distinctive of a given feature vector. In other words, those feature units that had the high discrimination value played a larger role in the face unit computation than those features that were less discriminating. For example, the fourth and fifth feature units distinguished the Joe prototype vector from the Bob and Tom prototype. Specifically, the fourth feature unit of the Joe prototype was 1 as compared to the fourth unit of Bob and Tom which was 0. Likewise, the fifth feature unit of Joe was 0 as compared to the fifth feature unit of Bob and Tom which was 1. Accordingly, the connections between the fourth and fifth feature units and the Joe face unit were weighted the most strongly. A strong positive weight connection between the fourth feature unit and the Joe face unit signaled strong evidence *for* the activation of the Joe face unit and a negative connection between the fifth feature unit and the Joe face unit signaled strong evidence *against* the activation of the Joe face unit. Thus, the distinctive feature information embodied in the connection weights of the model combined with the distinctive feature information found in the caricature vector produces the overall caricature advantage.

Implications and Limitations of the Model

The simulation results seem to be most consistent with the following interpretation of the caricature advantage: Faces are encoded in memory with respect to their distinctive properties. By emphasizing the same features in the stimulus that are distinctive in memory, caricature renderings can more strongly activate face representations than veridical drawings. Higher levels of activation lead to quicker access to face representations and faster recognition times.

The present simulation does not rule out the possibility that the recognition system encodes a distorted representation rather than a veridical representation. It only demonstrates that the extra computational step of distortion is not necessary to produce a caricature advantage. Therefore, the distinctive features hypothesis provides a more parsimonious account of the phenomenon. In fact, the Caricature Model would show a caricature advantage if the caricature vectors were learned instead of the veridical vectors. However, there may be additional reasons for treating caricature recognition as a special type of recognition rather than as a general model for face recognition. Theories accounting for the caricature advantage must also explain the ease with which the human recognition system can identify normal faces. If faces are stored as distorted caricatures, then normal, undistorted faces should be perceived as *anti-caricatures*; that is, a normal face would tend to de-emphasize its distinctive properties relative to the caricature representation. However, Rhodes et al. (1987) found that veridical drawings

of faces were better recognized than anti-caricatures indicating that normal faces are not treated as anti-caricatures.

Finally, a limitation of the current model is that the caricature advantage increases in proportion to the amount of exaggeration in the caricature vector (the β parameter in the caricature equation). This is not consistent with the empirical results of Rhodes et al. (1987) where they found that highly exaggerated drawings were not as quickly recognized as moderately exaggerated drawings. By adding an intermediate layer of hidden units to the model, it should be possible to simulate such non-linearities. Future simulation efforts will be directed toward investigating how modifications in the network's architecture might affect its behavior.

References

Barlett, J.C., Hurry, S., & Thorley, W. (1984). Typicality and familiarity of faces. *Memory and Cognition*, **12**, 219-228.

Brennan, S.E. (1985). The caricature generator. *Leonardo*, **18**, 170-178.

Gibson, J.J. (1973). On the concept of formless invariants in visual perception. *Leonardo*. **6**, 3.

Light, L.L., Kayra-Stuart, F. &Hollander, S. (1979). Recognition memory for typical and unusual faces. *Journal of Experimental Psychology: Human Learning and Memory*, **5**, 212-228.

Mauro, R., & Kubovy, M. (1988). Caricature and face recognition. Unpublished manuscript.

McClelland, J.L. & Rumelhart, D.E (1986). A distributed model of human learning and memory. In D.E., J.L. McClelland, & the PDP research group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Volume II.* Cambridge, MA: Bradford Books.

Rhodes, G., Brennan, S., & Carey, S. (1987). Identification and ratings of caricatures: Implications for mental representations of faces. *Cognitive Psychology*, **19**, 473-497.

Rumelhart, D.E., Hinton, G.E, & McClelland, J.L. (1986). A general framework for parallel distributed processing. In D.E., J.L. McClelland, & the PDP research group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition. Volume I.* Cambridge, MA: Bradford Books.

Winograd, E. (1981). Elaboration and distinctiveness in memory for faces. *Journal of Experimental Psychology: Human Learning and Memory*, **7**, 45-49.

Equilateral Triangles: A Challenge for Connectionist Vision

Subutai Ahmad[1] and Stephen Omohundro

International Computer Science Institute, Berkeley, CA.

ahmad@icsi.berkeley.edu, om@icsi.berkeley.edu

ABSTRACT

In this paper we explore the problem of dynamically computing visual relations in a connectionist system. The task of detecting equilateral triangles from clusters of points is used to test our architecture. We argue that this is a difficult task for traditional feed-forward architectures although it is a simple task for people. Our solution implements a biologically inspired network which uses an efficient focus of attention mechanism and cluster detectors to sequentially extract the locations of the vertices.

INTRODUCTION

Consider the visual task of determining whether a set of three point clusters form an equilateral triangle. People are very good at solving this kind of problem, but a connectionist implementation is not obvious. The difficulties posed by this problem are common to a wide variety of visual tasks and so we have used it as a touchstone against which to test visual neural architectures. In this paper we describe some of the fundamental operations it seems to require, biologically plausible neural implementations of those operations, and a computer simulation which combines them into a complete system.

The most straightforward visual neural representations assign a distinct unit to each visual pattern that must be classified. Unfortunately, the space of possible triangles is much too large for this kind of approach to be biologically possible. The optic nerve consists of about a million fibers from each eye, and so we consider square images which are a thousand pixels on a side. Since each of the three vertices can occupy any of these pixels, the total number of possible triangles in such an image is about $1000^6=10^{18}$. A brute force representation would require about a million times as many neurons as we have in our entire brain for just this one task. Just restricting the units to represent the set of equilateral triangles would still require about 10^{12} units.

If coarse coded representations are used, these numbers can be reduced somewhat, but many more examples will still be needed to learn to properly classify equilateral triangles than is biologically possible. The spatial relationships which define equilateralness will have to be discovered for each position, scale, and orientation of the triangle. Techniques have been proposed for introducing translation and rotational invariance into networks (Giles *et. al.*, 1987) which eliminate the need to include feature detectors at every location. Unfortunately these methods require that every unit have a large (quadratic) number of connections with complicated weight linkages between them. Furthermore, positional information is lost in these representations - one cannot retrieve the location and orientation of the objects in the image.

These difficulties would disappear if we could represent a triangle directly by the real valued coordinates of its vertices, say in the activations of 6 units. In this representation it is easy to construct units which compute the distance between a pair of points. With this setup, the network would only need to learn a classification function based on distances between points. This is a significantly easier task - the kind of task that simple backpropagation networks have been successful at. The main difficulty, of course, is to transform the repre-

1. The first author is also a graduate student in the Department of Computer Science at the University of Illinois at Urbana-Champaign.

sentation from a set of pixel values to a set of vertex coordinates.

Ullman (1987) has argued that many high-level visual tasks may be implemented as "visual routines". The idea is that a high-level system solves a visual task by choosing and combining a set of primitive visual operations. We will see that this framework is well-suited to our task, but that a plausible connectionist implementation is non-trivial. Later sections describe our solution to the equilateral triangle task with primitives for detecting and remembering the locations of clusters of points based on a focus of attention mechanism.

EVIDENCE FOR SEQUENTIAL VISUAL PROCESSING IN THE BRAIN

There is a large body of psychology literature which supports Ullman's idea of sequential visual processes. One of the best examples is the work by Treisman and her colleagues (Treisman & Gormican, 1988). Their work suggests that certain simple visual tasks are performed by people in parallel (response time is independent of the number of objects in the image) whereas other tasks require serial processing (response time is linear in the number of objects). Jolicoeur *et al.* (1986) have provided further evidence that people use sequential processes for certain visual tasks. They find that the time to report whether two stimuli lie on the same curve increases linearly with the distance between them along the curve. The presentation time is too short for saccadic eye movements.

There have also been recent neurophysiological results suggesting that neurons can dynamically change their response properties. Moran & Desimone (1985) report evidence that the sizes and locations of the receptive fields of certain neurons in the monkey visual cortex (Area V4) change with the task that the animal is trying to accomplish. This kind of behavior is suggestive of selective attention mechanisms which allow the processing of a high level system to be devoted to different regions of sensory input at different times.

NEURALLY PLAUSIBLE MECHANISMS FOR SEQUENTIAL VISUAL PROCESSING

Both of these lines of evidence suggest that serial processes play an important role in visual processing. (Ullman, 1984) suggests some specific processes which might be useful. Among them are primitives for deciding which portions of the image are relevant, selecting out these sections, and storing their locations for later processing. Given these primitives, a possible solution to our triangle problem is a system which sequentially selects and stores the location of each of the three clusters. Once the coordinates of the triangle are directly available, the distance between vertices can be explicitly computed. With this information the learning becomes trivial the network just has to learn the equality function with three inputs. In the next few sections we will describe a connectionist implementation of a fast focus of attention mechanism as well as mechanisms for detecting and storing cluster locations.

Locally Tuned Receptive Fields

In this section we describe a mechanism by which linear threshold units can give a localized response in a feature space. The following fact is exploited: if one maps the points in $\Re^{n-1}$ onto the paraboloid defined by $z = \sum_{i=1}^{n-1} x_i^2$, then the intersection of a hyperplane in $\Re^n$ with this paraboloid projects onto a sphere in $\Re^{n-1}$.

Thus there is a mapping between planes in $\Re^n$ and spheres in $\Re^{n-1}$. To select a set of points which lie within a sphere in some space one just has to project the points onto the paraboloid and slice it with the plane corresponding to the sphere. Points which lie "beneath" the plane are within the sphere. Figure 1(a) illustrates this for $\Re^2$. Notice that the computation of a threshold unit is exactly that of deciding on which side of a hyperplane an input point lies. To encode circular receptive fields with threshold units, you just need to include an extra input: the sum of the squares of all the other inputs. An equation of the form:

$$-(\sum w_i x_i + \sum x_i^2 + const) > 0 \qquad (1)$$

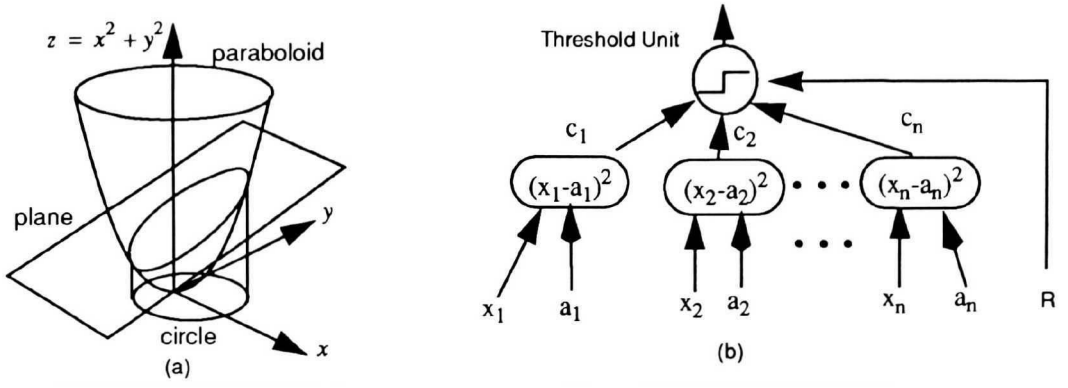


Figure 1(a) The plane intersects the paraboloid in a curve which projects to a circle. (b) Architecture of threshold unit computing the intersection shown in (a).The x_i's and a_i's are inputs to the system. The c_i's are weights denoting the stretch along each axis.

will be positive if $\dot{x}$ lies within a spherical volume determined by the weights and constant.

The above method creates circular receptive fields with hard boundaries. A smooth boundary with a flat top may be obtained by using a sigmoid instead of a threshold function. The steepness of the sigmoid (its gain) will then control the steepness of the receptive field boundary. Non-circular receptive fields are also possible by changing the nature of the non-linearity. Elliptical receptive fields may be obtained by using the paraboloid $z = \sum_{i=1}^{n-1} c_i x_i^2$ where c_i denotes the amount of stretching along each axis. In principle arbitrary shapes can be obtained by appropriately choosing the non-linearity.

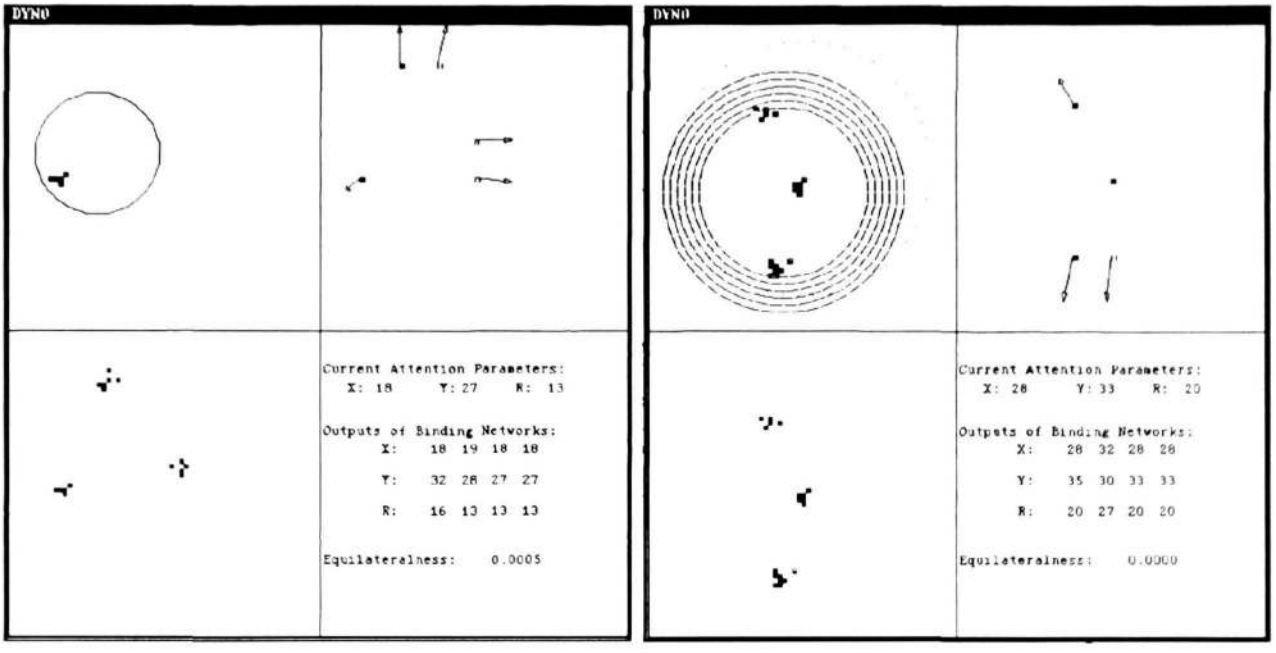
Dynamic Receptive Fields

In addition to being able to select a portion of the input space, we need the ability to shift the location and size of the receptive field around quickly in response to changing demands. In our model there are basically two ways of doing this. The first method involves changing the slope of the hyperplane. In Figure 1(a) note that as the slope increases the center of the projected circle will shift to the right. For any sphere it is possible to compute the coefficients of the hyperplane which produces that sphere. In a threshold unit, changing the slope of the hyperplane corresponds to changing the weights of the inputs. One of these units could eventually learn the correct position of its receptive field. However the time scale for weight changes is too slow for dynamic computation.

Another way to alter the sphere is to shift the paraboloid itself, by computing $z = \sum_{i=1}^{n-1} c_i (x_i - a_i)^2 + r$. This moves the paraboloid a distance a_i along dimension i (changing the location of sphere) and a distance r along the z-axis (changing the radius of the sphere). If the a_i's and r are available as input then the receptive field can be changed an arbitrary amount in one time step. Figure 1(b) shows how such a unit would be configured. For each input dimension there is a sub-unit which computes the square. ((Suarez & Koch, 1989) present a neurally plausible mechanism for computing a quadratic.) The outputs of these units are fed into a threshold (or sigmoid) unit. The net effect is that the threshold unit will respond only when the input vector $\dot{x}$ lies within the spherical receptive field determined by $\dot{a}$ and r.

Focus Of Attention With Value Coded Units

So far we have assumed an n-dimensional input space that is encoded as n analog signals. In our triangle task however, we have to implement a circular patch in a 2-dimensional retina. The units in this representa-



(a) (b)

Figure 2 Examples of the system behavior for two different 64x64 images. In both displays the lower left quadrant shows the image; the upper left quadrant shows the output of the gate units. The error vectors are displayed on the upper right and the outputs of various units are shown in the lower right. (a) shows a snapshot of the system with the focus of attention near one of the vertices. (b) shows the dynamic scaling behavior as the focus tries tofit the cluster of points within it.

tion are laid out on a flat sheet, with each unit explicitly encoding a point in the space.

To create dynamic receptive fields here, we construct a gating layer, with one "gate unit" per input unit. Each gate unit receives three global inputs: A_x, A_y, and A_r representing the parameters of the focus of attention. The gate units are basically the same as the localized units described above with one small modification: x_1 and x_2 are fixed for each gate unit. To encode this we have two separate connections from the input unit to the gate unit. The weights of these connections are fixed as x_1 and x_2. If the input unit fires, the threshold unit will fire only when its input unit is within the circle determined by A_x, A_y, and A_r. The effect is a layer of units which filters the input image according to a global control signal. The system can select any circular portion of the image in one time step. This is quite different from Mozer's implementation (Mozer, 1988) where the network has to iteratively settle on a solution, or Chapman's implementation (Chapman, 1990) which takes logarithmic time per selection. The hardware required to implement this is minimal: 8 extra connections per input unit. It is also fairly easy to extend our mechanism to allow foci of different shapes once the parameters are available.

Figure 2 shows the graphical output of our simulator. In each display, the lower left quadrant displays the current input image (3 point clusters). The upper left quadrant shows the output of the gate units. The circle shows the focus of attention represented by the 3 parameters. In Figure 2 (a) note that only the activity within the focus is allowed to propagate.

Deciding Where To Focus ... And How To Get There.

We need a mechanism to cause the focus of attention to sequentially fixate on the interesting portions of the image. There are two different cases to consider: 1) there is already a cluster within the focus of attention, and 2) the clusters are outside the focus. In the first case our system fixates on the center of mass of the cluster of points and wraps the focus of attention around the cluster. Once this is accomplished, the parameters of the focus of attention provide an accurate estimate of the location and size of the cluster. To deal with the

second case we have a system which receives input directly from the image and provides a rough estimate of the location of the next cluster to visit. These two mechanisms are described in the following sections.

Clusters Within The Focus Of Attention

The center of mass of a cluster, (C_x, C_y), is the average of the x and y coordinates of the active points:

$$C_x = \frac{\sum_i X(i)\, a_i}{\sum_i a_i} \text{ and } C_y = \frac{\sum_i Y(i)\, a_i}{\sum_i a_i} \tag{2}$$

where $X(i)$ and $Y(i)$ denote the x and y coordinates of the i'th unit and a_i denotes its activity. $\sum a_i$ can be computed by a unit which receives input from all gate units with a weight of 1. To compute $\sum X(i)\, a_i$ and $\sum Y(i)\, a_i$ we include two units with links to every gate unit. The weights from the i'th unit to each of these two units are $X(i)$ and $Y(i)$, respectively. A weighted sum of the incoming activity computes the appropriate value. The sums are divided by $\sum a_i$ to calculate C_x and C_y as the values of two units. These values are fed into two other units which compute the difference between the cluster center and the attention parameters: $(C_x - A_x, C_y - A_y)$. Units representing A_x and A_y receive as input this difference as well as their own output. By computing the sum of all their inputs, these units keep the field of attention continually centered on the cluster of points within the focus.

To get an estimate of the size of the cluster we include a scaling unit which continually adjusts the size of the focus of attention to match the size of the set of points within it. The rule is that as long as $\sum a_i$ remains constant, the scaling unit decreases A_r by a small amount. If the sum decreases, indicating that the scale has become too small, the unit increases A_r slightly and stops.

Figure 2 (b) shows the focus of attention changing to adjust to the cluster within it. The dotted circle represents the initial location. The set of concentric bands show successive steps as the focus of attention decreases and shifts its location to fit the cluster inside.

Clusters Outside The Focus Of Attention

The mechanism described above continually fine tunes A_x, A_y, and A_r to match the cluster of points within the focus but does not give us a way to fixate on clusters outside the focus. To do this we include a coarse grid of units which receives input directly from a circular patch in the image. At each grid location there are three outputs to consider. The first two outputs encode the difference between center of mass of the cluster of points within their receptive fields and the point (A_x, A_y). The third output is simply the number of active points within its receptive field, passed through a sigmoid. Thus each grid location encodes an "error" vector for adjusting the focus of attention, and a confidence value from 0 to 1 indicating the importance of the cluster. These error vectors are continually updated to compensate for changes in A_x and A_y. This error vector representation was inspired by the mechanism in the monkey superior colliculus for controlling eye saccades as reported by (Sparks, 1986).

The upper right quadrant of Figure 2 (a) and (b) shows the outputs of the error units. Each arrow represents the error vector at that location. The shaded square represents the confidence value - the darker the square the higher the confidence. (Only those locations whose confidence value is higher than 0.2 is displayed.)

To sequentially process each cluster in the image the system has to repeatedly select the largest confidence value, inhibit the corresponding unit, and send the error vector to the system controlling A_x and A_y. We are currently investigating several mechanisms for choosing the largest value. One could construct a winner-take-all network with competing confidence units such that the system settles into a state where only one

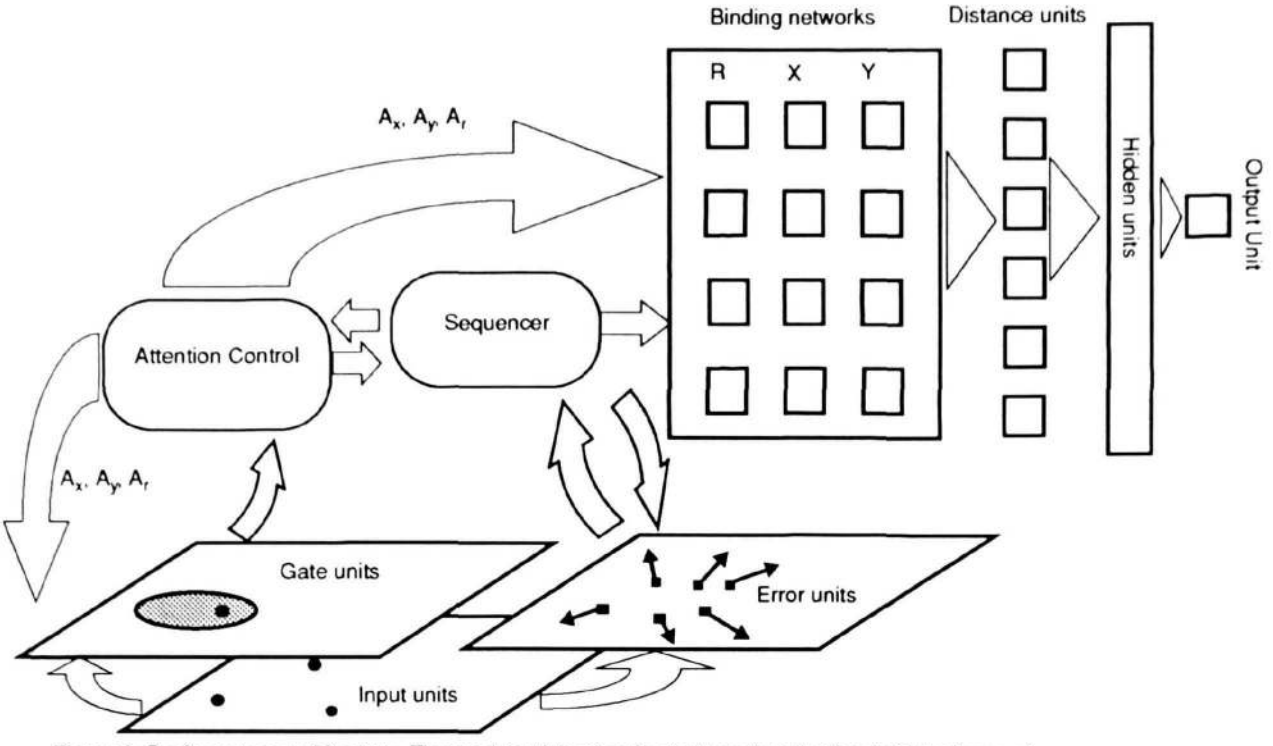


Figure 3. Basic system architecture. The module "Attention Control" continually tries to fit the focus of attention to the points within it. The "Sequencer" updates the focus of attention to visit all the clusters in sequence and also sends the control signals to the binding network to store the successive locations.

unit is active. However these networks can take a relatively long time to settle. It is also quite difficult to find the correct set of inhibitory weights to create a robust winner-take-all network. Another alternative is to construct a log-depth network of threshold units to explicitly compute the maximum, however this is equally unappealing. We currently assume a max finding unit but are studying a temporal representation which makes this computation biologically plausible.

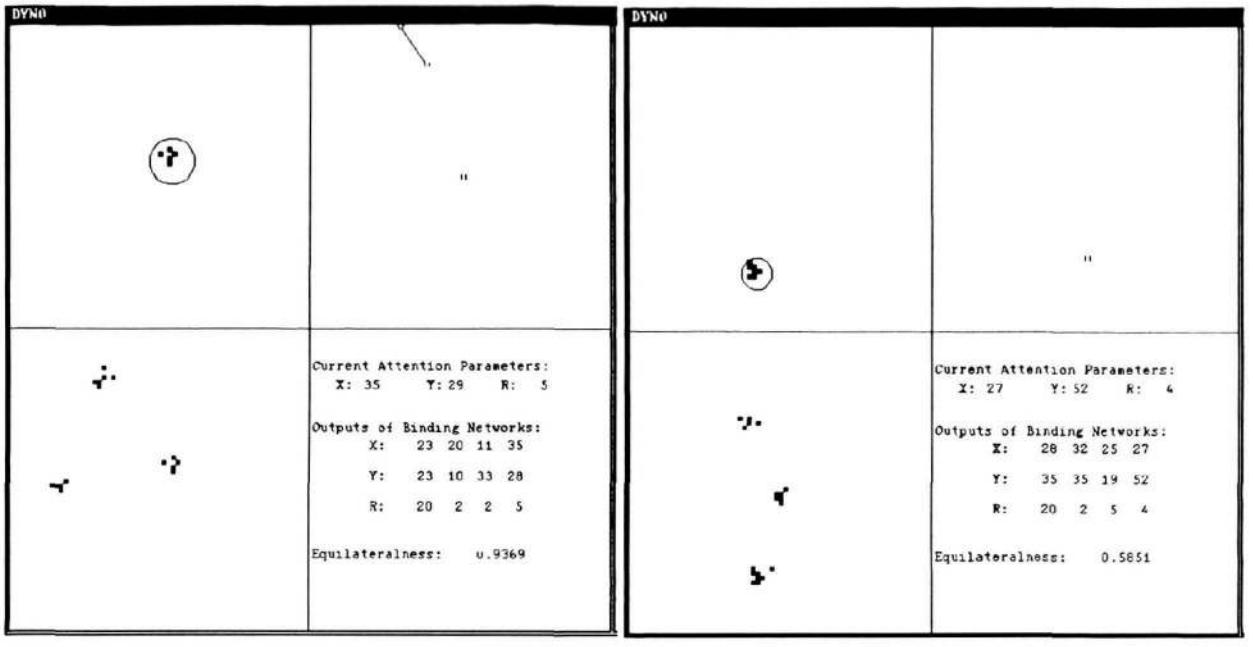
Storing Locations

As the network visits each vertex it should store the values of A_x, A_y, and A_r whenever the focus of attention stabilizes. We accomplish this with small recurrent networks (3 units) for each value that needs to be stored. Each of these networks continually tracks a particular unit (one of A_x, A_y, and A_r) until a control signal is sent, whereupon it freezes the output to be the current value of the unit. This is done by including a hidden unit which receives a strong inhibitory link from its control unit. When the control is off, it computes the difference between the value of the assigned unit and the current output and sends it to the output unit. An excitatory link from the control unit to itself ensures that once the control unit has fired, it stays on, preventing further adjustments. Three of these "binding networks" are used for each set of parameters that are stored.

SIMULATION

Figure 3 shows a schematic of the whole architecture. The module which controls the attention field is an autonomous network that continually attempts to wrap the focus around the points within its field of view. The sequencer waits until the focus of attention has stabilized and then does two things: it transmits a control signal to the binding networks to store the current parameters and updates the focus of attention to the location of the next cluster.

Assuming that the system starts with a focus of attention covering the entire image plane, the network first wraps the focus around the triangle and then sequentially visits and stores the locations of the three vertices.



(a) (b)

Figure 4. The result after parsing the two images shown in Figure 2.

Once this is done, the first set of bindings encode the position and scale of the triangle. The other nine bindings encode the positions and scales of the three vertices in the order that they were processed. A set of distance units then explicitly computes the six possible distances between the four stored locations. A standard feedforward network with one layer of hidden units is used to compute the final output. Space limitations prevent a more detailed explanation of the control structure. It is worth noting however that every aspect of it is implemented with units computing simple functions (except for the max function as described above). Details on the exact set up can be found in (Ahmad & Omohundro, 1990).

For the simulations in this paper we used images with 64x64 pixels. We generated a training set consisting of random triangles (approximately 50% of which were equilateral) with Gaussian noise added around each vertex. For each triangle the focus of attention was initialized to cover the entire image plane. The system was allowed to run until 4 control signals were generated. The outputs of the distance units were then used as inputs to train the backprop network. The teacher signal used was $1 - \frac{|l_1 - l_2| + |l_2 - l_3| + |l_1 - l_3|}{l_1 + l_2 + l_3}$, where l_i is the length of the i'th side. This is a function which is 1 for equilateral triangles and degrades gradually to 0 as the triangles deviate from equilateralness. With a training set of 100 triangles the network score was consistently greater than 0.9 for equilateral triangles. The specifics of the learning are not crucial to this paper, however note that the number of training examples needed would *not* increase if we increased the image size.

Figures 4 (a) and (b) show the state of the network after parsing the two triangles in Figure 2. The system correctly classified the left triangle as being equilateral and the right one as not being equilateral. The outputs of the binding networks show the vertex coordinates (in pixels) that were discovered by the network. The current implementation requires about 50 seconds on a Sun 4 to extract the coordinates of one triangle. By far the majority of this time is taken up by the scaling process.[1] This is because in our simulation we decrease the scale by a small constant at each time step. Starting off with a large focus of attention, a large number of steps may be necessary before the scale matches the cluster. We are investigating an implemen-

1. We don't actually need the scaling part for the triangle task however a general purpose cluster detector should have the ability to extract cluster sizes.

tation in which the scale shrinks continuously at a time scale faster than the interspike interval time of individual neurons.

DISCUSSION

It is interesting to speculate as to how mechanisms like the ones we have described would fit into a general purpose vision system. The library of primitives must be expanded to handle more features of realistic images. We will need primitives for intelligent processing of curves, regions, and shapes. Another issue is the large number of different representations that are formed in parallel in the brain. A mechanism for integrating the information available in these representations will be necessary. Since the focus of attention mechanism we described can be implemented in any topological space, in principle it should provide a good way of isolating exactly the subsets that are relevant.

Another major issue that we have not addressed is how to compile these visual primitives to accomplish a dynamically specified task. Such a system would presumably need a language for specifying the tasks. To translate the specification of the task into the appropriate primitives requires that the intermediate representations of the visual primitives and the descriptive language should be very similar. (Feldman *et. al.*, 1990) discusses some of these issues in the context of a novel approach to language acquisition.

In conclusion, the main point of this paper has been to demonstrate neurally plausible mechanisms for performing sequential visual computations. There is evidence from psychology and neurophysiology that biological organisms actually implement such routines at an early processing level. We have described an efficient implementation of various primitives within a connectionist framework, and have used them to extract image properties that are otherwise extremely inefficient to represent.

REFERENCES

Ahmad, S., & Omohundro, S. (1990). A Connectionist System for Extracting the Locations of Point Clusters. Technical Report TR-90-011, International Computer Science Institute, Berkeley, CA.

Chapman, D. (1990). *Instruction Use in Situated Activity.* Ph.D. Thesis, Massachusetts Institute of Technology.

Feldman, J.A., Lakoff, G., Stolcke, A., & Weber, S.H. (1990). Miniature Language Acquisition: A Touchstone for Cognitive Science. Submitted to the 12th Annual Conference of the Cognitive Science Society, MIT, July 1990.

Giles, C.L., Griffin, R.D., & Maxwell, T. (1987). Encoding Geometric Invariances in Higher Order Neural Networks. In "Advances in Neural Information Processing", David Touretzky, Ed. Morgan Kaufmann.

Jolicoeur, P., Ullman, S., & Mackay, M. (1986). Curve tracing: A possible basic operation in the perception of spatial relations. *Memory and Cognition*, **14** (2), 129-140.

Minsky, M. & Papert, S. (1969). *Perceptrons: An Introduction to Computational Geometry.* MIT Press, Cambridge, MA.

Moran, J. & Desimone, R. (1985). Selective Attention Gates Visual Processing in the Extrastriate Cortex. *Science*, **229**, March 1985.

Mozer, M. (1988). A Connectionist Model of Selective Attention in Visual Perception. University of Toronto Technical Report CRG-TR-88-4

Sparks, D. L. (1986). Translation of Sensory Signals into Commands for Control of Saccadic Eye Movements: Role of Primate Superior Colliculus, *Physiological Reviews*, **66** (1).

Suarez, H., & Koch, C. (1989). Linking Linear Threshold Units with Quadratic Models of Motion Perception. *Neural Computation*, **1** (3), pp 318-320.

Treisman, A., & Gormican, S. (1988). Feature Analysis in Early Vision: Evidence From Search Asymmetries. *Psychological Review*, **95** (1), pp 15-48.

Ullman, S. (1984) Visual Routines. *Cognition*, **18**, pp 97-159.

The Computation of Emotion in Facial Expression Using the Jepson & Richards Perception Theory

Brent C.J. Britton

The Media Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

brent@media-lab.media.mit.edu

Abstract

Facial expressions are vital communicators of emotions, and it is in partial response to these expressions that we innately and accurately discern the emotional states of those around us. This paper identifies the activatable facial features that form the language of emotional expression in the face, and the set of emotions that each such feature tends to express. Finally, it is shown how the **fault lattice perception theory** [6] can be used to compute the emotion being registered on a face, given the configuration of the salient features. It is posited that the ability of a computer to make such interpretations would significantly enhance human-computer interaction.

1 Introduction

The ability of the human face to express a wide range of emotions is well supported.[2, 4, 3, 1] The face is stimulus and response in one, a remarkably effective and versatile communicator; human facial muscles are sufficently complex to produce more than a thousand different facial appearances.[2] Our impressions of the facial expressions on those with whom we interact is so seemingly innate, and the communication of emotional information so useful, that a specialized cognitive system may have evolved which is capable of discriminating facial expressions and making inferences about them.[5] Constructing a similar system for use at the computer-human interface seems beneficial.

It is currently possible to communicate with a properly equipped computer system through the use of speech, hand gestures, and eye movements (in addition to, of course, keyboard and mouse input), all of which contribute to a natural, human-like communication environment. Bestowing upon the computer the ability to interpret facial expressions would enrich human-computer communication even more, as it would allow the computer to become sensitive to the emotional state of its human user.

Humans supplement their understanding of facial expressions with contextual knowledge gained through interaction and observation. It is rare that we are expected to interpret a person's emotional state strictly by the examination of his or her face alone. We have the luxury of being able to assimilate other subtle yet important clues such as a person's tone of voice and body movements, as

well as any external stimuli that may affect a person's emotional state. The scenario for the computer in this paper, however, is far more simplistic. Here, the computer is assigned the task of determining the user's emotional state (perhaps within the underlying context of ascertaining the user's response to information being provided by the computer) by examining a "snapshot" of the user's face. Adequate vision processing technology is assumed.

The **fault lattice perception theory** [6] serves as the theoretical basis for development. The theory seems particularly well suited to this problem, since, as we shall see, emotional expression is a systematic result of describable facial features (giving rise to a rich set of premises).

2 The Language of Emotions

The manifold facial expressions can be decomposed into gross categories corresponding to a handful of general, primary emotions that people are capable of readily perceiving. The primary emotions can be labeled more or less adequately by the seven terms **happiness, surprise, fear, sadness, anger, disgust**, and **interest**.[1] The face itself can be separated into its constituent parts (brows, eyes, lips, etc.) the relative arrangements of which are likely to be similar and consistent within an emotion category.

2.1 Features

What are the features of a given facial configuration that lead observers to perceive the expression of a particular emotion? To answer this, we must first develop a taxonomy of facial features and their possible states.[1]

F1: **brow**[2] can be b1: neutral, b2: raised, b3: lowered

F2: **eyes** can be e1: open (neutral), e2: wide open, e3: slit, e4: closed, e5: tightly closed

F3: **nose** can be n1: neutral, n2: wrinkled, n3: flared (nostrils)

F4: **lips** can be l1: open or wide open, l2: pushed out, l3: curved up, l4: curved down, l5: closed (neutral), l6: tightly closed

F5: **jaw** can be j1: neutral, j2: pushed forward, j3: dropped

The feature knowledge (**F1** through **F5**) serves as a core axiom when the fault lattice perception theory is applied. Note that each of the five features has a neutral state.

The reader may wonder why the teeth are not included as a significant facial feature. As it happens, I originally included them, but soon discovered their state on the face to be just as accurately described by the lips and jaw. Furthermore, it turns out that for nearly every one of the emotions (below), it is acceptable for the teeth to be either open or closed, making them a rather useless indicator!

2.2 Expressions

The aggregate states of the five facial features form the overall facial expression. Let us now see how each of the seven primary emotions is typically expressed by the feature states.[3] It is from these descriptions that useful premises arise for use in the fault lattice perception theory.

happiness: brow raised, **eyes** neutral or slit[4], **nose** neutral, **lips** curved up, **jaw** neutral.

[1] The list of features is an adaptation of the **Action Units** of the **Facial Action Coding System** given in [3]. Action Units, however, are anatomical *processes*, whereas the features I list are static *states*.

[2] To simplify matters, I use the term **brow** in reference to *both* eyebrows, with the assumption that they are incapable of moving independently of each other.

[3] These descriptions are adapted from [4] and [3].

[4] The eyes often appear slit in a happy expression as a result of the intensity of the smile.

surprise: **brow** raised, **eyes** wide open, **nose** neutral, **lips** open or wide open, **jaw** neutral or dropped.

fear: **brow** neutral or lowered, **eyes** closed or tightly closed, **nose** wrinkled, **lips** open or wide open, **jaw** neutral.

sadness: **brow** neutral or lowered, **eyes** neutral or slit, **nose** neutral, **lips** closed, curved down and pushed out, **jaw** neutral or dropped.

anger: **brow** lowered, **eyes** wide open or slit, **nose** wrinkled or flared, **lips** tightly closed, or wide open, **jaw** pushed forward.

disgust: **brow** lowered, **eyes** slit or closed, **nose** wrinkled and flared, **lips** curved down and pushed out, **jaw** neutral or dropped.

interest: **brow** neutral or lowered, **eyes** neutral or slit, **nose** neutral, **lips** closed, **jaw** neutral.

A neutral face is one in which all features are in their neutral states. As it happens, the expression of **interest** is composed almost entirely of neutral facial features, thereby making it a good candidate for the "default" perception, the interpretation chosen in the event that none other is stronger. The practical result is that a neutral face may be perceived as expressing **interest**, and this is likely to have no effect whatsoever on the interaction between human and computer — it should be fair to assume that if the user is expressing no obvious emotion about the information being provided by the computer, then he or she is at the very least *interested.* It is for this reason that the concept of the neutral expression can safely be disregarded in favor of **interest** when ordering interpretations. However, the computer is likely to require an initial "snapshot" of the user's neutral face against which to compare later images to determine the degree of change in the various feature states.

2.2.1 Categories

The seven primary emotions are grouped into two categories, **pleasant** (happiness, interest, surprise[5]) and **unpleasant** (fear, sadness, anger, disgust). Certain feature states are exclusive to pleasant emotions:

b2: raised brow

l3: upwardly curved lips

while certain others are exclusive to unpleasant ones:

e4: closed eyes

e5: tightly closed eyes

n2: wrinkled nose

n3: flared nose

l2: outwardly pushed lips

l4: downwardly curved lips

l6: tightly closed lips

j2: forwardly pushed jaw

Thus it is often possible to narrow down the choices for a given expression by checking for the presence of the exclusive features. Even though more direct mappings (in the form of the feature state descriptions given above) are already at our disposal, they do not always lead to the confident impression of a single primary emotion. The exclusive features, when present, allow us to choose between interpretations from the different categories.

[5] Whether or not **surprise** can be said strictly to be a pleasant emotion is debatable (see, for example, [4], as well as the Conclusions section of this paper). It does, however, seem reasonable in the context of the human-computer interface under discussion here.

2.2.2 Blends

There is some controversy surrounding the concept of legal emotional **blends** in facial expressions.[2, 1, 3] Surely, it is possible to have two emotions blended in the same expression — you can be both happy and surprised, for example — and, it seems, blends may even be more common than single emotions in expressions.[2]

However, it is not clear precisely which emotions can or cannot be blended, owing to the fact that few of the seven primary emotions have, in the remaining six, a direct opposite with which a blend can confidently be called illegal. Furthermore, the descriptions of blends can be fairly complex (is the brow of one emotion blended with the lips of another to be considered identical to the lips of the first blended with the brow of the second?).[2] It does not seem wise, therefore, to attempt to enumerate in detail all the possible emotional blends or the facial features that contribute to these expressions. What *does* seem plausible, though, is that emotional blends across the pleasant/unpleasant boundary are extremely unlikely — a blend of **happiness** and **sadness** is an obvious example — and it is this notion that we shall adopt as a core axiom.

3 Applying the Theory

To invoke the fault lattice perception theory we must first enumerate the infallible core axioms and the fallible premises.

3.1 Core Axioms (not fallible)

Much of the foregoing is embodied in the core axioms. Interpretations at odds with the axioms are regarded as inconsistent.

A1: Accept the feature knowledge **F1** through **F5**.

A2: A neutral expression is interchangeable with an expression of interest.

A3: Accept the pleasant/unpleasant categorization of the emotions, and the exclusive features.

A4: A viewed facial expression is a truthful representation of one of the seven emotions listed above, or a blend.

A5: A pleasant emotion cannot be blended with an unpleasant emotion.

3.2 Premises (fallible)

Each of the premises consists of a feature state paired with the emotion or emotions that it seems most likely to indicate in all cases.[6] The premises are constructed so that all seven primary emotions are indicated by at least one of the states of some facial feature. That is, for each facial feature (brow, eyes, etc.), there is at least one state (raised, lowered, etc.) that indicates a given emotion. Although this may seem to be an unwarranted and fairly large reduction in the amount of known information, it must be noted that some of the facial feature states listed in the descriptions of the seven primary emotions in the previous section are somewhat tenuous and do not always apply as typically as others, so their inclusion here could lead to errors. Also, humans seem to find it quite easy to discern the emotions given only a partial view of the face, indicating that only certain facial features are necessary to create the expression.[3] Thus it seems proper simply to choose, for each facial feature state, the emotion or emotions with which the feature state is most closely identified.

B1: A neutral brow indicates interest or sadness.

B2: A raised brow indicates surprise or happiness.

B3: A lowered brow indicates anger, fear, or disgust.

E1: Neutral eyes indicate interest or happiness.

[6] Again, these choices are adapted from [4] and [3], and are formulated in part by my own subjective opinions.

E2: Wide open eyes indicate surprise.

E3: Slit eyes indicate anger or sadness.

E4: Closed eyes indicate disgust.

E5: Tightly closed eyes indicate fear.

N1: A neutral nose indicates happiness, interest, surprise or sadness.

N2: A wrinkled nose indicates anger, disgust or fear.

N3: A flared nose indicates anger.

L1: Open or wide open lips indicate surprise or fear.

L2: Pushed out lips indicate disgust.

L3: Upwardly curved lips indicate happiness.

L4: Downwardly curved lips indicate sadness.

L5: Closed lips indicate interest.

L6: Tightly closed lips indicate anger.

J1: Neutral jaw indicates happiness, interest, or fear.

J2: Forwardly pushed jaw indicates anger.

J3: Dropped jaw indicates surprise, disgust, or sadness.

3.3 Ordering Interpretations

The list of premises may seem formidable, but only a handful of them (five) are in use at a time. Any feature (**b1** - **j3**) present on the face under examination activates the corresponding premise (**B1** - **J3**). Premises based on other feature states are irrelevant. To accept a premise means to assert, obviously enough, that the feature state indicates one of the named emotions, and to fault a premise means to assert that the feature state does not indicate any of the named emotions. The acceptance or faulting of the activated premises in various combinations leads to interpretations of various emotions. The label (emotion) assigned to an interpretation is the emotion or emotions (if any) common to all accepted premises, and not forbidden by the faulted premises.



Figure 1: Expression of surprise.

Consider the photograph in Figure 1. The (somewhat exaggerated) expression on this man's face is, according to its source, universally perceived as **surprise**.[2] Thus, if our premises are sound, the fault lattice perception theory should yield **surprise** as a local maximum in the lattice.

To begin, the face is examined for the presence of key features. Even the most superficial glance at the expression in Figure 1 reveals the following features:

b2: raised brow

e2: wide open eyes

n1: neutral nose

l1: wide open lips

j3: dropped jaw

Consequently, the following premises are activated:

B2: A raised brow indicates surprise or happiness.

E2: Wide open eyes indicate surprise.

N1: A neutral nose indicates happiness, interest, surprise or sadness.

L1: Open or wide open lips indicate surprise or fear.

J3: Dropped jaw indicates surprise, disgust, or sadness.

Let us proceed by attempting to formulate a plausible interpretation while assuming that all of our assumptions about the world are incorrect — that is, when all five of the activated premises are faulted.

Faulting premise **B2** means that a raised brow (or, more specifically, the raised brow in Figure 1) indicates *neither* **surprise** *nor* **happiness**; faulting **E2** means that wide open eyes *do not* indicate **surprise**; and so on for the remaining active premises. Thus, when all of the premises are faulted, the interpretation can be neither **surprise, happiness, interest, sadness, fear**, nor **disgust**. Consequently, invalidating our worldly assumptions leads us to the conclusion that the face in Figure 1 bears an expression of **anger**, the only emotion not explicitly forbidden by the faulted premises!

We are bound, however, by core axioms **A3** and **A5** which tell us infallibly that certain facial features (in this case, b2: raised brow) are exclusive to pleasant emotions, or, conversely, that they can never appear on expressions of unpleasant emotions (in this case, **anger**). The result, then, is that **anger** is an inconsistent intrepretation. This gives us at least a little confidence in the soundness of our premises, so let us now see what happens when we accept just one of them.

Accepting **B2** means that the raised brow indicates **surprise** or **happiness**. Neither of these is yet a plausible interpretation, however; each is ruled out by one or more of the remaining (faulted) premises. More precisely, **happiness** is inconsistent with **N1** (which, being faulted, forbids the **happiness** interpretation, among others), and **surprise** is inconsistent with *all* the remaining premises (which, being faulted, summarily forbid the **surprise** interpretation). Thus, accepting only **B2** leads to an inconsistent interpretation to which we cannot assign a label. It should also be obvious that accepting only **E2** leads to the very same sort of inconsistent interpretation.

But what happens when we accept only **N1**? This means that the neutral nose indicates **surprise, happiness, interest** or **sadness**, one of which, namely **interest**, is not forbidden by any of the remaining (faulted) premises, nor is it inconsistent with any of the core axioms. Thus we are able to assign a label of **interest** to the interpretation in which **N1** is the only accepted premise.

Though continuing to enumerate the rest of the 32 possible combinations of accepted and faulted premises would be an instructive exercise, it soon becomes clear that the process is unnecessarily tedious. It is straightforward enough simply to examine the sets of emotions indicated by the activated premises and decide by inspection which combinations are interesting and lead to plausible interpretations. For example, it is easy to see that accepting **B2** and **N1** together leads to the interpretation of **happiness**, since this emotion is indicated by both premises and forbidden by none of the others. The fault lattice for the expression in Figure 1 which we have been studying appears in Figure 2. Many of the unlabeled nodes are not shown. Note that, as was hoped, **surprise** does indeed turn out to be a maximal node.

The expression in Figure 1 was nice to us; it was exaggerated enough to trigger the feature states beyond any doubt. However, in the majority of

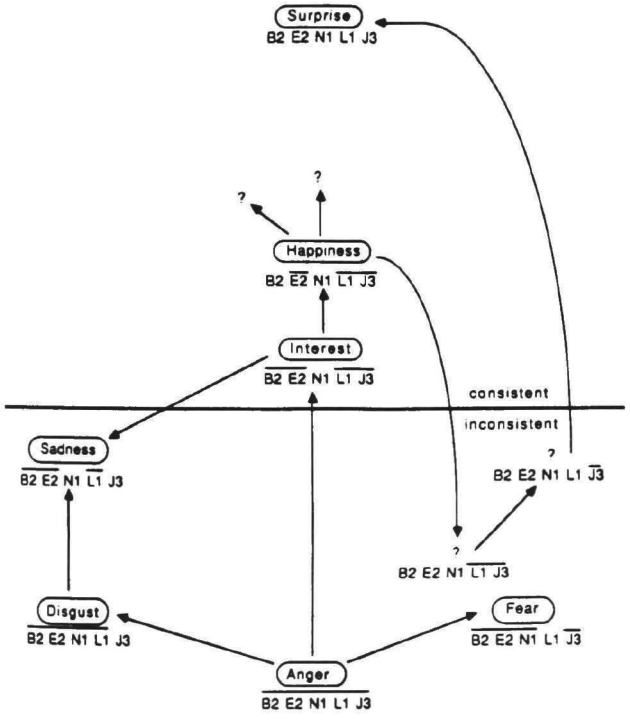


Figure 2: Fault lattice for the expression in Figure 1.



Figure 3: Expression of intense anger.

the faces we see (unlike those typically used in subjective studies) the facial features are quite subtle. We have assumed that our computer has on hand a "snapshot" of the user's neutral face against which to compare later "snapshots" of the same user to determine the amount of change in the features, and we can further assume that this comparison process is carried out at the resolution necessary to detect such subtleties. For the purposes of this paper, however, all we have to go on are our own judgements, because we are not always provided with neutral "snapshots" for comparison. We must therefore take care not to let our judgements of the facial features be biased by our expectations.

Again, there was little doubt about our choices for the expression in Figure 1, but consider the expression in Figure 3, described by its source as intense anger.[2] The features on this face are quite subtle, but it seems safe to say that the brow is neutral, the eyes are slit, the lips are tightly closed, and the jaw is neutral. The nose, however, could

be neutral, but it *could* be flared, too. It's just not possible to tell from this one photograph. Carrying out the analysis as if the nose is neutral (using premise **N1**) results in a lattice (not shown) in which **anger** is not the maximal node, but is a *local* maximum. The lattice in which the nose is taken to be flared is shown in Figure 5, and in it, **anger** is the maximal consistent node. The case in which no premises are faulted is unlabeled, since there is no emotion common to all premises. Furthermore, the lips are taken to be tightly closed, and since this feature is exclusive to unpleasant emotions, then interpretations of pleasant emotions are inconsistent by core axioms **A3** and **A5**. For example, in the case where **J1** is the only accepted premise, the interpretation is a *don't know*, a toss up between **fear** and **happiness**. Since **happiness** is a pleasant emotion, however, it is inconsistent, so this node can be labeled **fear**. The same situation exists in the five-fault case with **disgust** and **surprise**.

4 Conclusions

The premises the we have adopted for use in this exercise seem to be fairly sound, and have generated lattices in which the expected interpretations are at least local maxima. A more robust analysis might consider more complex premises that account for the meanings of certain *combinations* of facial feature states, or that use more granularity in the descriptions of the feature states themselves.

Although core axioms **A3** and **A5** seem to be exceptionally useful in preventing obviously incorrect interpretations from becoming consistent perceptions, the notions of exclusive features and pleasant/unpleasant categorizations are in some ways too restrictive. More precisely, there are situations in which it does not seem wise to relegate **surprise** and **interest** strictly to the pleasant category, thus making them unable to be blended with unpleasant emotions. Indeed, these two emotions can almost be thought of as *measures* for the

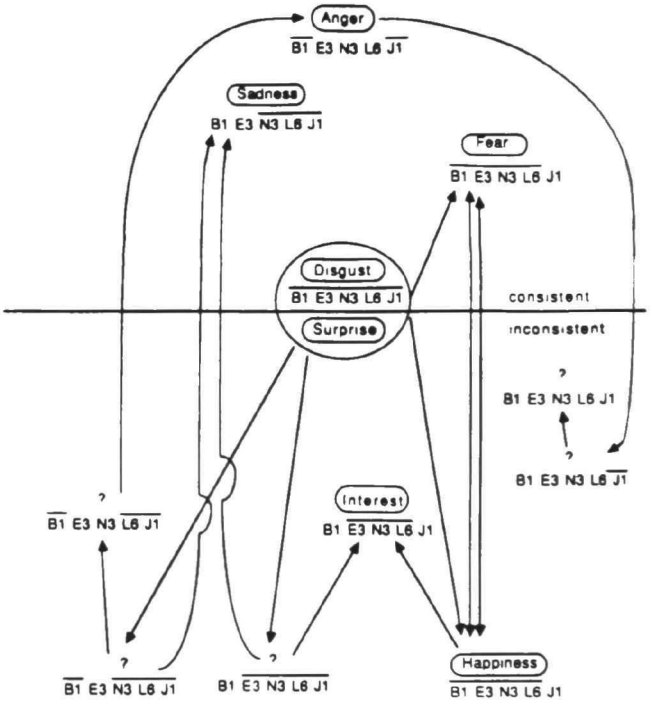


Figure 4: Fault lattice for the expression in Figure 3 when the nose is taken to be flared.

remaining five, levels of saturation along a continuum, always present to some degree in every face. In this respect, greater precision may be required in the categorization of emotions, and the determination of legal blends.

It has been assumed throughout this paper that the computer system attempting to read the expression of its user is doing so without regard for any other forms of input, and without the use of any higher-level reasoning. It does seem entirely plausible, however, to program a system to make use of contextual clues such as what the user says, how the rest of the user's body moves, and the sort of reaction that the information being provided by the computer is *likely* to illicit based on general expectations, or a knowledge of how this particular user has reacted in similar situations in the past.

In summary, the analysis of facial expressions is certainly not an exact science, but the results shown here indicate that it is possible to correctly interpret emotion by examining the configuration of certain facial features. The decisive task is to compose a rich set of premises that accurately describe the relationships between the features and the emotions they indicate.

References

[1] Graham Davies et al., editors. *Perceiving and Remembering Faces.* Academic Press, London, 1981.

[2] Paul Ekman, editor. *Darwin and Facial Expression.* Academic Press, New York, 1973.

[3] Paul Ekman, editor. *Emotion in the Human Face.* Cambridge University Press, Cambridge, 1982.

[4] Hadyn Ellis et al., editors. *Aspects of Face Processing.* Martinus Nijoff, Dordrecht, 1986.

[5] Nancy L. Etcoff. The neuropsychology of emotional expression. MIT Center for Cognitive Science Occasional Paper number 31.

[6] A. Jepson and W. Richards. What is a perception? AI Version, 1989.

[7] Whitman Richards, editor. *Natural Computation.* MIT Press, Cambridge, Massachusetts, 1988.

[8] Clea Theresa Waite. The facial action control editor, face. Master's thesis, MIT Media Arts and Sciences Section, 1989.

Imagery and Problem Solving*

Yulin Qin
Herbert A. Simon
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

In this paper we discuss the role of imagery in understanding problems and the processes of using images to solve problems. On the basis of two experiments and computer simulation, we show how subjects, in solving a particular problem, form mental images to represent a changing physical state. By "running" and watching the mental image they can draw qualitative conclusions about the situation, then derive a quantitative equation to solve the problem.

1. Introduction

Imagery can integrate local information and make explicit the spatial relations among objects (Kosslyn, 1980; Larkin and Simon, 1987). Images can also be manipulated, for example, by mental rotation (Shepard and Metzler, 1971; Shepard and Cooper, 1982). In this paper we will describe two experiments, one in detail and the other briefly, that show how imagery can be used to represent a process. By "running" and watching the image, subjects draw qualitative conclusions that then allow them to derive the relevant equations that solve the problem.

2. Experiment 1

2.1 Instruction

In this experiment, the subjects were instructed as follows:

Image

This task will allow you to check your capacity for imaging.

Please try to see these events in your mind:
(1) There is a rod AB, i.e., one end of the rod is named A, the other end is named B; ok?
(2) This rod is moving, with velocity v, along its axis; ok?
(3) Now, try to see that a ray of light is emitted, with velocity c, from A towards B along the axis of the moving rod.
OK?
Based on your image, please derive the expression of the time which the light will spend when it travels from A to B. (Hint: distance = time * velocity)
[You can write down any formula while you derive the result, if you need to. BUT, please DO NOT draw any diagram. Watch the picture **in your mind!**]

2.2 Method

Fifteen subjects, members of a class in English as a Second Language, were run. The

* This research was supported by the Computer Science Division, Office of Naval Research, under contract number N00014-86-K-0678. Reproduction in whole or part is permitted for any purpose of the United States Government. Approved for public release; distribution unlimited.

subjects were foreign graduate students and visiting scholars at Carnegie Mellon University. The instructions were printed at the top of a sheet of 8 1/2 x 11 paper, leaving the lower half as the answer sheet. After the subjects had received the instructions, the experimenter read them aloud, sentence by sentence. When the subjects responded "Ok" for one sentence, the experimenter read the next one.

After reading the instructions, the experimenter told the subjects that they would be given five minutes to derive the equation called for by the instructions, and emphasized: "You can write down any formula while you derive the result, if you need to. BUT, please DO NOT draw any diagram. *Watch the picture in your mind!*" When five minutes had elapsed, the experimenter asked the subjects: "Please draw the picture in your mind on the other side of the paper."

After giving the correct equation to the subjects, and explaining the purpose of the experiment, the experimenter collected the answer sheets. Ten subjects returned their answer sheets.

2.3 Results

Figure 1 shows the diagrams drawn by the subjects, all of them reduced in scale by the same amount. We will comment on these drawings presently.

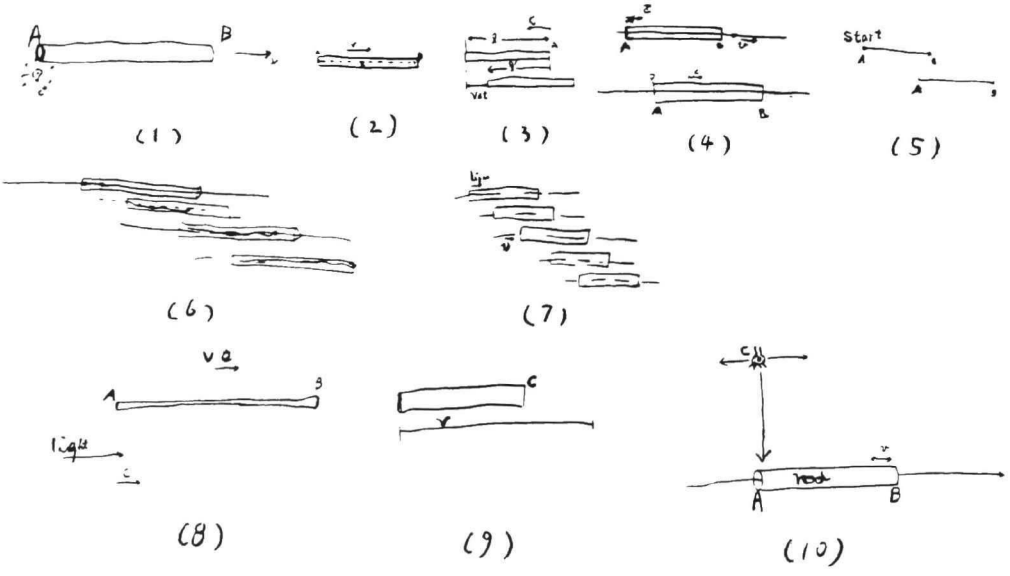


Figure 1.

The equation to be written down depends on how subjects choose the relative directions of c and v. If c and v are assumed to lie in the same direction, then the answer is $t = r_{AB} / (c-v)$, where r_{AB} is the length of the rod. If c and v are assumed to lie in opposite directions, then the answer is $t = r_{AB} / (c+v)$.Within 5 minutes, 5 of the 10 subjects, those drawing diagrams (1) to (5),

obtained a correct answer. Two other subjects, those drawing diagrams (6) and (7), wrote the (incorrect) equation: $t = r_{AB} / c$. The 3 subjects who drew diagrams (8) to (10), did not write down an equation.

As we have noted, the instructions were (intentionally) ambiguous in some respects, and the values of some variables were given only implicitly. For example:

a) In sentence 1, the direction of movement of the rod is not specified, nor whether the rod lies in the horizontal, vertical, or some other direction. It is not specified which end of the rod is A, and which B. Nor is there information about the length or shape of the rod.

b) Sentence 2 provides no information about the direction of the velocity v.

c) Both v and c are, implicitly, measured relative to a stationary frame of reference -- presumably the frame of the subject, watching the moving rod.

2.4 Individual Differences

To form their mental images, the subjects had to add information to resolve the ambiguities and make implicit values explicit. Only one subject asked about the direction of v while the experimenter was reading the instructions, and was told, "It depends on your choice."

From Figure 1, we see that there are large differences in the information added by different subjects, e.g., the length and shape of the rod, and the way of representing the light (2 subjects drew electric bulbs). However, much of the information was introduced in almost the same way by nearly all the subjects. For example:

a) All of the rods in the images were laid horizontally.

b) 9 of 10 subjects used the left end of the rod as A, and the right end as B.

c) 8 of 10 subjects pointed v and c in the same direction, from A to B; only one subject, Figure 1, (3), pointed v from B to A, but c from A to B as called for by the instructions. One subject, Figure 1, (10), drew vectors v and c in both directions.

There was less information about the consistency among subjects in making implicit values explicit. The 5 subjects who solved the problem assumed that v and c were to be measured relative to the subject's stationary frame. The two subjects who obtained the answer $t = r_{AB} / c$ might have chosen c relative to the position of the rod. The frames of reference used by the 3 subjects who did not write equations cannot be determined.

We see that the subjects are relatively consistent in the way in which they supply information, although there is considerable variety in the visual appearance of their diagrams. In Figure 1, we see that 5 diagrams depict the successive locations of the rod, giving us a clue as to how the images represent a moving process, and how, on this basis, the subjects can derive the quantitative equation. Notice that none of the subjects who failed to write equations showed the rod in more than one location, in contrast with 5 of the 7 who wrote an equation. Presently, we

will ask why two subjects who solved the problem did not depict the rod in more than one location, although doing so can be important to the reasoning.

2.6 Deriving the Equation from the Image

There are at least two routes by which the equation can be derived from the image. Let us assume that c and v are chosen with the same direction, from A to B. One route to the solution begins with the inference that the *net* velocity of the light relative to the rod is c - v. The other route rests on the inference that the light, relative to the stationary frame, must travel, in order to reach B from A., the distance $r_{AB} + vt$ -- that is the length of the rod, plus the distance (vt) the rod has traveled during the time of travel (t) of the light, so that $t = (r_{AB} + vt) / c$.

Of the 5 subjects who solved the problem, 2 used the former (relative velocity) route, while the other 3 used the latter (distance) route. In Figure 1, 1) and 2) are the drawings of the subjects in the velocity group, while 3), 4), and 5) are the drawings of the subjects in the distance group. The velocity group did not, but the distance group did, depict the rod in different locations. On the basis of the diagrams, we can infer that all the subjects noticed that while the light traveled from A to B the rod also moved. The subjects in the velocity group noted this by recording the velocities; the subjects in the distance group by recording the initial and final positions of the rod.

3. Discussion

From the fact that the velocity and distance groups of subjects recorded different kinds of images, and that at the same time, they used different lines of inference to the conclusion, we conclude that there is a close correspondence here between imagery and reasoning processes. If we assume that the reasoning takes place in a representation defined by the imagery, we see that the imagery determines what reasoning steps are available. (Cf. Paige and Simon, 1966.)

When the experimenter explained the answer, and tried to show the subjects how to use the relative velocities (the algebraically simpler procedure) to solve the problem, some subjects found it difficult to understand the reasoning. We, the authors, also find that it is easier to image the distance than the relative velocity.

The experimental evidence does not prove definitively that subjects chose the velocity route *because* they saw the relative velocities in their mental imagery. However that may be, using the relative velocities to solve the problem (perhaps because they had learned that concept in previous physics training) would direct their attention to the velocities, and would make the successive positions of the rod irrelevant. In contrast, the subjects in the distance group did form an image that depicted the traveling rod. However, they did not try to show motion, as such, but simply the positions of the rod at the two critical moments -- the time when the light was emitted, and the time it reached the end of the rod. In this as in many other problems (Paige and Simon,

1966), change can be represented, not as a continuous process, but simply in terms of the "before" and "after" states of the system.

The process we have sketched out above for the distance group in solving the problem, which seems consistent with the data, can be outlined as follows:

a) Setting the goal: to find the time required for the light to travel from A to B.

b) Forming and running the mental image, i.e., doing the mental experiment.

c) Observing the changed states of rod and light.

d) Noting that the distance the light travels is greater (or less) than r_{AB}.

e) Deriving the qualitative result that the time required is greater (or less) than r_{AB}/c.

f) Determining the quantitative value of the difference (vt) between r_{AB} and the distance the light traveled.

g) Deriving the equation: $t = r_{AB} / (c-v)$ (or, $t = r_{AB} / (c+v)$).

h) Checking the quantitative with the qualitative result.

A corresponding model, going from image to qualitative inferences and from qualitative inferences to quantitative equation, could be described for the subjects in the relative velocity group. We can gain further insight into the model for the distance group from the protocol of a subject taken in an earlier experience. We turn to this protocol next.

4. Experiment 2

This experiment was carried out in our laboratory. Subjects were asked to understand, with the help of their mental images, the first, kinematical, part of Einstein's 1905 paper: "On the Electrodynamics of Moving Bodies" (Einstein, 1905). This part is focused on the basic concepts and the deriving of the Lorentz transformation equations. Einstein derives one of the key equations as follows (with questions inserted by the experimenters in the reading material in bold face):

> We imagine further that at the two ends A and B of the rod, clocks are placed which synchronize with the clocks of the stationary system, that is to say that their indications correspond at any instant to the "time of the stationary system" at the place where they happen to be. These clocks are therefore "synchronous in the stationary system."
>
> **[Q : Following the instruction given in this part, try to form an image in your mind, describe this image to us, and then draw it.]**
>
> We imagine further that with each clock there is a moving observer, and that these observers apply to both clocks the criterion established in §1 for the synchronization of two clocks. Let a ray of light depart from A at the time t_A, let it be reflected at B at the time t_B, and reach A again at the time t'_A.
>
> **[Q : Following the instruction given in this part, try to form an image in your mind, describe this image to us, and then draw it.]**
>
> Taking into consideration the principle of the constancy of the velocity of light we find that
>
> $$t_B - t_A = r_{AB}/(c - v)$$
>
> and
>
> $$t'_A - t_B = r_{AB}/(c + v)$$
>
> where r_{AB} denotes the length of the moving rod -- measured in the stationary system.

[Q: Here are the equations about the characteristics of the image described in the reading material. Tell us how you would use the image to explain and justify these equations.]

We can see that the instructions of our Experiment 1 are a simplification of this part of the reading material from Experiment 2. The major difference is that in Experiment 1 the subjects were not given the final equation, but had to find it, while in Experiment 2, the equations were given to the subject. However, in the latter experiment, the subjects were also asked to explain the equations in terms of their mental images.

Protocols were obtained from 7 subjects, most of them undergraduate or graduate students in Electrical and Computer Engineering or in the School of Computer Science at Carnegie Mellon University. We will examine a single, particularly informative, protocol here, The protocol of S_g. S_g tried both the velocity and distance routes to a solution.

Form image

S_g read the material, formed an image and drew it , as shown in Figure 2.
[After he had finished his diagram, the diagram was taken away by the experimenter]

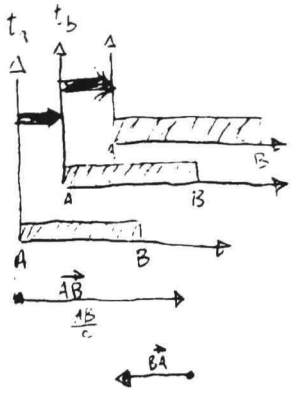


Figure 2.

Read equations

S read the equations.

Examine image

"the length of that ray and--Uhmm--we see that is the sum of observe the length of the rod and the amount that it moved because--Ahmm--the ray left--Ahaa--left from-Ahaa--the origin with respect to our stationary system and it moved along at a constant speed C-Ahmm"..."So that --Ahaa--that that is the--okay--and then you take take the length of that ray you divide that by c minus v and Oh boy! This is hard to visualize Uhmm."

Set goal.

"Uhmm--We said we want to find this quantity $t_B - t_A$ "

Find starting point, image and qualitative relation

"Uhmm and what do we know--Ahmm--about that?--Uhmm-- We know that it took okay the entire distance" "it had to travel the length of the rod--Ahmm--which is -- r_{AB}. Uhmm--let me see that it is the same but that's not enough because its moved-- Ahaa--the, the rod has moved you can see that by looking at the t_B frame and see that--Uhmm--its moved--during that that time that it traveled "

Gesture

His left hand and right hand moved toward right in parallel

Transfer equations, guided by the image

"Should I try and do this symbolically? Should I--Uhmm should I think of my picture here and--Uhmm--try and show how I can justify these equations? "
S copied the equations from the reading material and transferred them as follows:

$r_{AB} = (t_B - t_A)(c - v)$

$r_{AB} /(t_B - t_A) = c - v$

$r_{AB} = (t'_A - t_B)(c + v)$

$r_{AB} /(t'_A - t_B) = c + v$

Consider relative velocity

"Uhmm--and--Ahaa--ray travels at a speed c in respect to the stationery frame but the whole rod is moving with a positive velocity v so the ray is actually traveling with

respect to the moving frame at a velocity c - v --Uhmm--so--Uhmm--if we want to derive the time that it took for this ray to get from t_B or t_A to t_B when--Ahaa--let's see what this is parts of r_{AB} over c-v."

[E: Why are these c-v and c+v ?]

Do semi-quantitative reasoning	Total time times light velocity = length of the rod + some component, because the rod is moving. $c(t_B \quad t_A) = r'_{AB}$
Do quantitative reasoning	$r_{AB} = r'_{AB} - v(t_B - t_A)$ $r_{AB} = c(t_B - t_A) \quad v(t_B - t_A)$ $r_{AB}/(c - v) = (t_B - t_A)$

From S_g 's protocol, we can see that his process for deriving the equations was not as "neat" as our model. He used means-ends analysis, trying different methods. There was frequent alternation between forming and examming the mental image and engaging in reasoning. S obtained some clues from the form of the given equation, and was able to derive it by means of the relative velocities. But he could not explain clearly what he had done until he shifted to the distance route. Then he could explain more clearly, following closely the path of our model except for the insertion of a "semi-quantitative reasoning" episode in his protocol.

When a second subject, S_j, tried the relative velocity route for solving the problem, he imaged two trucks running in the same or opposite directions as an analog, a method of doing qualitative reasoning. Although there were many individual differences among the protocols in Experiment 2, the general processes revealed in them are essentially those of our models.

5. Computer Simulation

We are building a program in OPS5 to simulate this process, and have finished the qualitative and quantitative reasoning parts. The input of this program is information derived from the image.

The major qualitative reasoning production rules are the following:

If the direction of the light is the same as that of the rod then the distance the light travels will be larger than the length of the rod.

If the direction of the light is opposite to the rod then the distance the light travels will be less than the length of the rod.

If the direction of the light is the same as that of the rod then the distance the light travels will be the length of the rod plus the distance the rod moved.

If the direction of the light is opposite to the rod then the distance the light travels will be the length of the rod minus the distance the rod moved.

If the direction of the light is the same as that of the rod then the velocity of the light relative to the rod is less than that relative to the stationary system.

If the direction of the light is opposite to the rod then the velocity of the light relative to the rod is larger than that relative to the stationary system.

Figure 3 gives two examples of the input and output of the program. The "IMAGE" part is the input of the program. The remainder is the output. Part (1) is the simulation of the distance group , and part (2) is the simulation of the velocity group. The difference between the input of these two examples is only in the relative system for the velocity of the light.

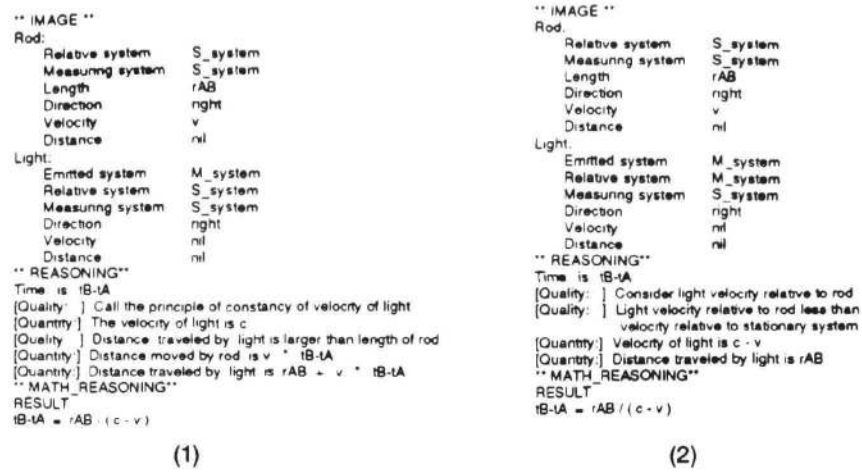


Figure 3.

6. Conclusion

From our two experiments, we can conclude:

1. For a simple physics problem involving relative motion, subjects can form mental images to represent a process of change in the physical state of a system. They can "run" the image to carry out mental experiments and watch the change of the physical states.

2. By watching the process, subjects can infer the qualitative relations among the physical variables, and draw qualitative conclusions.

3. Based on previous knowledge (e.g., distance equals rate times time), subjects can derive quantitative equations from the qualitative relations.

4. The process is not generally linear. There is a good deal of interaction among the processes of forming and observing the mental image and reasoning from it.

In the simple example of relative motion, we have seen that there may be more than one route to the solution, and that different routes may correspond to different mental images. In the example discussed here, the subjects did not have to change the concepts they already held, or construct new concepts. In other protocols we have observed such changes in concepts, and will discuss them in subsequent papers.

Reference

Einstein, A. (1905). On the Electrodynamics of moving bodies. in A. Einstein, et al. *The Principle of Relativity*. Dover Publications, 1923.

Kosslyn, S.M. (1980). *Image and Mind*. MA:Harvard University Press.

Larkin, J.H. and Simon, H.A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11(1), 65-100.

Paige, J. M. and Simon, H. A. (1966). Cognitive processes in solving algebra word problems, in B. Kleinmuntz (ed.), *Problem Solving*. New York: John Wiley & Sons.

Shepard, R. N. and Copper, L.A. (1982). *Mental Images and Their Transformations*. MA:MIT press

Shepard, R.N. and Metzler, J. (1971). Mental rotation of three-dimensional objects. *Science*,171, 701-703.

Psychological Simulation and Beyond *

Ian Pratt
Department of Computer Science,
University of Manchester,
Manchester, M13 9PL, UK
ipratt@uk.ac.man.cs.ux

Abstract

In this paper, we examine the suggestion that inferences about another person's state of mind can proceed by *simulation*. According to that suggestion, one performs such reasoning by *imagining* oneself in that person's state of mind, and *observing* the evolution of that imagined cognitive state. However, this simulation-based theory of psychological inference suffers from a number of limitations. In particular, whilst one can perhaps observe the probable *effects* of an given cognitive state by putting oneself in that state, one cannot thus observe its probable *causes*. The purpose of this paper is to propose a solution to this problem, within the spirit of the simulation-based theory of psychological inference. According to the *indexing thesis*, certain cognitive mechanisms required for non-psychological inference can be re-used for hypothesising psychological causes. The paper concludes by discussing some of the possible implications of the indexing thesis.

1 Psychological inference

Let us begin with an example. Suppose a company employee E is told by his colleague F to take some coffee to a certain office where a meeting has been taking place. Suppose further that E knows that the meeting has just broken up. Then if E is minimally intelligent and co-operative, he will not blindly obey the instruction, but will pass his information on to F. In doing so, E has inferred the following: (i) that F *wants* coffee to be taken to the office; (ii) that F wants this because F has the *goal* of giving the people in the meeting coffee; and (iii) that F *believes* (falsely) that the people at the meeting are still in the office. We call inferences involving the *goals*, *plans*, *beliefs*, *etc.* of other agents *psychological inferences*.

In this paper, we examine the suggestion that psychological inference can proceed by *simulation*. According to that suggestion, one reasons about the state of mind of another

*This work is supported in part by the Joint Councils' Research Initiative in Cognitive Science. MRC grant number SPG8920254.

person by *imagining* oneself in that person's situation, and *observing* the evolution of one's imagined cognitive state. However, this simulation-based theory of psychological inference suffers from a number of problems. The purpose of this paper is to examine one of those problems, and to suggest a solution within the spirit of the original theory.

2 Background

Within artificial intelligence, research into psychological inference has fallen into two main traditions. The first, arising from the work of the Yale school on story-understanding (e.g. Wilensky[14]) focuses primarily on *plan-attribution*. On this approach, inferring the plan behind an agent's actions involves the repeated application of special rules (sometimes called 'reverse planning rules') which allow the system to pass from those actions to the goals that they facilitate or achieve. Recent work in this tradition includes that of Allen[1] and Litman and Allen[10] on plan-attribution in human dialogue.

The second tradition of psychological inference within artificial intelligence uses *epistemic logics* to reason about belief and knowledge, whilst tending to neglect goals, plans, intentions, etc. (see Konolige[9] for a survey). These epistemic logics themselves fall into two distinct classes, corresponding to two competing philosophical accounts of belief and knowledge. *Propositional logics*, following the work of Hintikka[7], take the objects of belief to be propositions—equivalently, sets of possible worlds. In thus characterising belief as a relation between a believer and a set of possible worlds, it is natural for such logics to idealise the rationality of agents by taking belief to be both consistent and closed under logical consequence[1]. The second class of epistemic logics, the *sentential logics*, take the objects of beliefs to be sentence-like entities, sentences, as it were, in the agent's language of thought. Sentential logics appear to be more promising for modelling sub-rational agents, in particular, agents whose beliefs may be inconsistent or not logically closed. Included in the category of sententialist logics, and of particular relevance to this paper, is the work of Haas[6], who, following Creary[4], takes psychological inference to proceed by *simulation*.

3 The simulation thesis

The simulation-based theory of psychological inference adopts the thesis that psychological inference is sometimes a matter of simulation (hereinafter, the *simulation thesis*). The idea can be illustrated with an example taken from chess. In deciding on a move in a chess game, I must reason about which threats will be obvious and which difficult to see until it is too late. According to the simulation thesis, to decide whether a given threat will be obvious

[1]Some epistemic logicians have worked on overcoming these idealisations (see, for example Levesque[11] and Fagin and Halpern[5]). Other researchers, however—particularly within the philosophical community—regard consistency and closure as desirable properties for any reconstruction of the notions of belief and knowledge, not as mere idealisations whose sole purpose is to render epistemic logics simple and perspicuous. See, for example, Stalnaker[13].

to my opponent, I imagine myself in his position, and observe how easily I can, within that imagined cognitive predicament, discover the threat for myself. That is, I imagine myself knowing what my opponent knows (the positions of the pieces on the board, but not the other player's immediate plans), and desiring what my opponent desires (that he should win), and I *simulate* the processes that may be presumed to be occurring in his mind.

Notice that the fact that I actually know the purpose of my chess gambit is nothing to the point. In imagining myself in my opponent's position, I temporarily suspend this knowledge, working only from what I think my opponent knows. In doing so, I automatically simulate the time- and space-constraints under which my opponent is labouring, for the simple reason that, in performing the simulation, *I* am labouring under similar constraints. Thus, the simulation-based theory of psychological inference prides itself on the natural way in which it yields conclusions about agents whose reasoning powers are sub-optimal: predictions of sub-optimality due to limitations of time, space and knowledge arise naturally from the fact that whoever is doing the prediction will be labouring under such constraints himself.

I suggest that psychological reasoning by psychological simulation occurs often in everyday situations. Why might I worry that my friend may doubt that I like him? Because I have, unavoidably, missed several recent appointments with him, uttered comments which, on reflection, might have been misinterpreted, and so forth. I know that the comments might have been misinterpreted because I can imagine hearing them (and can imagine having the particular concerns and interests with which I know my friend to be currently preoccupied), whereupon the offense-giving interpretations, which had not occurred to me before, now become obvious. I know that these interpretations, together with the missed appointments and so forth, might cause my friend to doubt my loyalty because, as I imagine having experienced the things I believe him to have experienced, that thought occurs to me.

Again, the fact that I actually know that I still like my friend, or that I can devise a clever argument which would convince my friend that his doubts are unjustified, are nothing to the point. It is enough that, in contemplating the situation from my friend's point of view, I can experience coming to a particular conclusion. To repeat: the simulation-based theory of psychological inference prides itself on the natural way in which it yields conclusions about agents whose reasoning powers are sub-optimal.

In reasoning about another person's reasoning by imagining oneself in that person's position, one is exploiting the fact that, in one's own brain, one has an analogue model of another person's brain: the ability to imagine oneself in alternative cognitive predicaments is the means by which one can compensate for the differences between one's own thinking and that of the agent one is simulating. Notice that, in using psychological simulation to predict that another person might well fail to see a threat in chess, or fail to make an inference, or otherwise engage in some seductive but erroneous piece of thinking, one need not have any sort of *theory* of how that person thinks. Such knowledge is unnecessary because psychological inference can simply trade on the fact that one person's cognitive mechanisms are really very much like another's. For instance, when I predict my opponent's likely responses to a chess gambit, I use many of those cognitive mechanisms which enable me to reason non-psychologically about chess. Whatever cognitive mechanisms that allow

me to scan a board for threats and opportunities, compute the values of exchanges of pieces and so forth, now find alternative employment: they enable me to make inferences about how my opponent will or may think in a given situation. I do not need extra axioms or rules which describe how I do these things, or which tell me how long I will take to do them: such axioms and rules would at best amount to a pointless duplication.

4 A problem with the simulation-based theory

The simulation-based theory of psychological inference has a long history, going back at least to Hobbes[8]. More recently, interest in simulation was re-kindled by Kenneth Craik[3], and similar ideas have been discussed in a variety of contexts in artificial intelligence. (For a survey, see Pratt[12].) Although simple in outline, however, the simulation thesis quickly leads to considerable difficulties which must be addressed if the simulation-based theory is to be developed.

Here is one such difficulty. The chess example illustrates how, if one imagines oneself in a given cognitive predicament, one can observe the way in which that predicament will evolve. But whilst one can thus observe what *later* states a given cognitive predicament would normally *give rise to* (i.e. what *effects* it would normally have), one cannot observe what *earlier* states would normally *have given rise to* that cognitive predicament (i.e. what *causes* it would normally have). Yet many of the psychological inferences we want to perform involve inferring not the *effects* of given psychological states, but their *causes*. So it is with the example of section 1: when E attributes to F the goal of giving the people in the meeting coffee, E is making an inference (to us, an obvious inference) about what earlier psychological state *caused* F's decision to ask E to take the coffee to the office. And it is difficult to see how such an inference could proceed by simulation: by definition, psychological simulation can only run forward in time.

To be sure, if one has a limited number of specific hypotheses about the psychological causes of an action, one may test those hypotheses by simulation: simply by putting oneself into each of the hypothesised psychological states in turn, and seeing if the results are consistent with one's current view of the agent one is simulating. But simulation can be of no help in *generating* a suitably constrained set of hypotheses in the first place. The inability of simulation to generate hypotheses about the causes of psychological states constitutes a major limitation of the simulation-based view of psychological inference. Apparently, something more than psychological simulation is required for effective psychological inference.

5 The indexing thesis

I come now to examine a partial solution to the problem of generating hypotheses about the causes of psychological states, a solution which I take to preserve the spirit of the simulation-based theory. The current status of the solution is purely that of a proposal. However, in the next section, I explore some of its implications with a view to its empirical testability.

The basic idea can be illustrated by comparing the following two inferential tasks.

> (1) Imagine yourself in a room with some very high shelves and a large wooden chest, and consider the action of dragging the chest over to the shelves. If you now ask what potentially useful goal this action might enable you to achieve, the answer will be obvious: that of reaching the shelves.
>
> (2) Imagine yourself in the same room, but now suppose that someone tells you to put the chest beneath the shelves. If you now ask yourself why this person intends that the chest should be placed there, the answer will again be obvious: presumably, that person wants to be able to reach the shelves.

Prima facie, inferences (1) and (2) are very different. In inference (1), one thinks up a useful goal given a condition (or given an action to achieve that condition) which makes it possible to realise that goal; in inference (2), by contrast, one hypothesizes the cause, in terms of a higher-level goal, of someone's intention to achieve a given state. Inference (1) is not at all psychological (there is no reasoning about anyone's reasoning); inference (2), by contrast, is a centrally about the psychological cause of a psychological state. Let us call the first mode of reasoning, in which the problem is simply to think of a goal G that a given condition P enables or facilitates, *goal-activation*; let us call the second mode of reasoning, in which the problem is to hypothesise a possible goal G as a cause of an agent S's intention to bring about P, *goal-attribution*. Goal-activation has a forward-chaining character. One starts with some state P and thinks up a goal that P will help achieve. Goal-attribution, by contrast, is often a matter of *reasoning about* S's backward-chaining inference. One supposes S to have started with some goal G and, as a result, to have adopted some plan which involves achieving P; the problem is: given P, solve for G.

But in spite of their differences, these two modes of reasoning have a similar character. The crux of the similarity is this: in both cases, the reasoner must rely on some cognitive mechanism whereby goals are *indexed* by conditions or actions that enable or facilitate them. For example, in both inferences (1) and (2) above, the reasoner must be able to think up the goal of reaching the shelves by considering the state of affairs in which the chest is beneath the shelves. Now, a plausible thesis is that, fundamentally, the same goal-indexing mechanisms are used in *goal-attribution* as are used in corresponding cases of *goal-activation*. In particular, the same indexing mechanisms are used in inferences (1) and (2) above. Let us call this thesis the *indexing thesis*.

When one is faced with the problem of looking for the cause of an agent's intention to achieve some state of affairs P, and when one is considering the possibility that that intention may have arisen through a process of backward-chaining planning from some goal G, then, in seeking possible candidates for G, one performs forward-chaining reasoning starting from P, looking for sensible goals that P might help achieve. Put loosely, one might say that, according to the indexing thesis, the cognitive mechanisms one uses in (non-psychological) forward-chaining inference get re-used in reasoning about the backward-chaining inference of other agents. Though the indexing thesis and simulation thesis are logically independent of each other, they are nevertheless in much the same spirit. For, according to both

theses, cognitive mechanisms needed anyway for wholly non-psychological inference find alternative employment in psychological inference. However, whilst the simulation thesis is about how one infers the probable (psychological) *effects* of a given psychological state, the indexing thesis is about how one hypothesises some possible (psychological) *causes* of a given psychological state. Thus, the indexing thesis complements the simulation thesis.

6 Implications

We can get some idea of the implications of the indexing thesis by considering the effect on goal-activation and goal-attribution of one's knowledge of whether various conditions obtain that affect the achievability of certain goals. These implications suggest that the indexing thesis is empirically testable, as I shall explain.

It is important to realise that, in goal-activation, a new (or newly perceived) condition P may suggest a goal G even if G is not directly attainable as things stand. Consider, for example, a variant of inference (1), in which the shelves can only be reached by placing a stool on top of the chest. In that case, the action of pushing the chest to beneath the shelves may still suggest the goal of reaching the shelves, even though achieving that goal may require an extra action (putting the stool on top of the chest). Clearly, if goal-activation is to be useful, a situation P (or an action that achieves it) must sometimes suggest a goal G even if G requires the achieving of further preconditions which are not as yet satisfied. (Indeed, such preconditions may turn out to be unsatisfiable.) On the other hand, our knowledge of what conditions obtain must constrain the the process of goal-activation to *some* extent: otherwise there is no limit to the hopeless goals that any condition P might suggest. Ideally, the less likely G is to be achievable given P, the less likely a reasoner should be to think of G when he considers what goals P might facilitate. Thus, in goal-activation, the indexing of a goal G by a precondition P should be sensitive to one's beliefs concerning the various other preconditions of G[2].

Now, the indexing thesis states that the same goal-indexing mechanisms underly both goal-activation and goal-attribution. But we just argued that, in goal-activation, the indexing of a goal G by a precondition P should be sensitive to one's beliefs concerning the various other preconditions of G. If so, then, we should expect that in goal-*attribution*, the indexing of a goal G by a precondition P will be similarly sensitive to one's beliefs concerning the various other preconditions of G.

The question arises as to whether this consequence of the indexing thesis desirable. Now, in many goal-attribution tasks (e.g., the example of section 1), one must be prepared to attribute to an agent a goal G which only makes sense on the supposition that that agent has a false belief about some precondition for achieving G. Indeed, such cases of goal-attribution are often essential for detecting these false beliefs. But then, one needs a mechanism for thinking up a goal G given a precondition P, even if, according to *one's own* knowledge,

[2]For brevity's sake I speak of the *preconditions of goal G* rather than the *preconditions of actions which will achieve goal G*.

goal G has some unsatisfied (or even unsatisfiable) preconditions. On the other hand, one should not be prepared to attribute *any* collection of false beliefs to the agent: for then there is no limit to the hopeless goals one can consistently attribute to him. Thus, effective goal-attribution demands that one use a goal-indexing mechanism which is sensitive in some way to one's beliefs about the satisfiability of the various preconditions of the relevant goals.

Yet the suggestion contained in the indexing thesis that the same goal-indexing mechanisms are responsible for *both* one's ability to perform sensible goal-attribution in the face of false beliefs on the part of the agent one is reasoning about, *and* one's ability to perform sensible goal-activation in the face of unrealised pre-conditions for the goals that one suggests. The question now arises as to whether goal attribution and goal activation really do call for exactly the *same* pattern of sensitivity to unrealised preconditions. Or, is it rather the case that the optimal goal-indexing strategy for goal-activation is different from that for goal-attribution? And if—as indeed seems likely—goal activation and goal attribution would *ideally* require different indexing mechanisms, is it feasible to construct a compromise between the two: something that serves adequately, if not optimally, for both tasks? And if so, is that what people actually do? That is: do human subjects exhibit the corresponding patterns of sensitivity to unrealised preconditions in both goal-activation and goal-attribution that the indexing thesis predicts?

As remarked above, the status of the indexing thesis is, at present, that of a suggestion for extending the simulation based theory of psychological inference. What is required, if this suggestion is to be fleshed out and evaluated as a psychological hypothesis, is a detailed investigation of the foregoing questions. The purpose of this paper has merely been to point the way to these investigations.

7 Summary

My point of departure in this paper was the simulation-based theory of psychological inference. According to that theory, one can reason about an agent's state of mind by imagining oneself in that agent's cognitive predicament and simulating his thought processes. The simulation thesis embodies the attractive idea that the same cognitive mechanisms that the reasoner uses for ordinary (non-psychological) inference get re-used in psychological inference. Thus, a reasoner does not need a *theory* of how psychological states interact; he can simply trade on the presumed similarity between himself and the agent he is simulating.

However, the simulation thesis suffers from a number of limitations, among them, the problem that simulation cannot generate hypotheses as to the *causes* of a given psychological state. The purpose of this paper is to propose the *indexing thesis* as a possible, partial solution to this problem. According to the indexing thesis, hypotheses about the psychological causes of another agent's present actions can sometimes be generated with the aid of cognitive mechanisms which are required for non-psychological goal-activation tasks. Although logically independent of the simulation-thesis, the indexing thesis also embodies the attractive idea that certain cognitive mechanisms that the reasoner uses for non-psychological inference (goal-activation) get re-used for psychological inference (goal-attribution). The

cognitive mechanisms in question are mechanisms for indexing goals by conditions or actions that will achieve or help achieve those goals. We concluded by outlining some implications of the indexing thesis[3].

References

[1] Allen, James F.: "Recognising Intentions from Natural Language Utterances", in Brady, M. and Berwick, Robert C.: *Computational Models of Discourse*, Cambridge, MA: MIT Press (1984), pp. 107–166.

[2] Allen, James F. and Perrault, C. Raymond: "Analysing Intention in Utterances", *Artificial Intelligence*, **15**, 3 (1980), pp.143–178

[3] Craik, Kenneth: *The Nature of Explanation*, Cambridge: Cambridge University Press (1943).

[4] Creary, Lewis G: "Propositional Attitudes: Fregean Representation and Simulative Reasoning", *Proceedings, IJCAI-79*, Tokyo (1979).

[5] Fagin, Ronald and Joseph Halpern: "Belief, awareness and limited reasoning", *Artificial Intelligence* **34**, 1 (1987), pp. 34–76.

[6] Haas, Andrew R.: "A Syntactic Theory of Belief and Action", *Artificial Intelligence* **28**, 3 (1986), pp. 245–292.

[7] Hintikka, J.: *Knowledge and Belief*, Ithaca, NY: Cornell University Press (1962).

[8] Hobbes: *Leviathan*, (ed. C.B. Macpherson) Harmondsworth, Middlesex: Penguin Books (1968).

[9] Konolige, Kurt: *A Deduction Model of Belief*, London: Pitman (1986).

[10] Litman, D. and Allen, James F.: "A Plan-Based Recognition Model for Subdialogues in Conversation", *Cognitive Science* vol.11, no.2 (1987), pp. 163–200.

[11] Levesque, H.J: "A logic of implicit and explicit belief", *Proceedings, AAAI-84*, Austin, Texas (1984).

[12] Pratt, Ian: "Psychological Inference, Constitutive Rationality and Logical Closure", in *Vancouver Studies in Cognitive Science, vol. 1: Information, Language and Cognition*, Vancouver, BC: University of British Columbia Press (1990, forthcoming).

[13] Stalnaker, Robert C: *Inquiry*, Cambridge, Massachusetts: MIT Press (1984).

[14] Wilensky, R.: *Planning and Understanding*, Reading, MA, Addison Wesley (1983).

[3] I would like to thank Ivan Leudar for discussion of some of the ideas in this paper.

Formal Models for Imaginal Deduction

George W. Furnas
Bell Communicaitons Research

Abstract

Systems with inherently spatial primitives have advantages over traditional sentential ones for representing spatial structure. The question is how such representations might be used in reasoning. This paper explores a simple kind of *deductive* reasoning where picture-like entities, instead of symbol-strings, are given first-class status. It is based on a model of deduction as the composition of mappings between sets, and allows generalized notions of unification and binding, which in turn permit the definition of various formal, "imaginal" deduction systems. The axioms and rules of inference are all pictures or fundamentally picture-based, and are used to derive pictorial "theorems". After sketching the generalized theory needed, several possible strategies are mentioned, and a prototype, the BITPICT computation system, is described in some detail.

1. Introduction

The role of imagery in reasoning, human or artificial, remains unsettled. In cognitive psychology Shepard[1], Kosslyn[2], and others have argued that imaginal representations exist and certain geometric transformations of them (rotations, scanning, zooming) can be used to answer spatial questions. Explicitly addressing inference, Lindsay[3] has argued that images are important because of their ability to implement geometric constraints automatically and that this is what distinguishes imaginal reasoning from "calculus-plus-proof-procedure," deductive systems. A different view is offered here: that imaginal *deductive* inference is possible, and specially empowered by the ability of images to implement spatial constraints.

Consider a simple example from a conventional deduction system, shown in Figure 1(a). This might be part of a system which manipulates arithmetic expressions while preserving their arithmetic value. The figure shows Axiom 1, representing the commutative law of addition, and Axiom 2, representing the *right*-zero-additive-identity law. These two axioms are put together in a deductive chain to yield the indicated Theorem, the *left*-zero-additive identity. In this algebra example, the rules link expressions that are one-dimensional arrangements of symbols. The goal here is to have the rules link picture-like objects. A domain that yields a fairly intuitive example is in classical Euclidean geometry proofs, illustrated in Figure 1(b). Axiom 1, expressed in English, would be something like, "If one angle of a transversal of two parallel lines is designated as being of interest, the corresponding second angle may be designated as its equal." Here, however, the axiom is meant to stand on its own as a statement in a formal system, with the rule linking the left-hand picture to the right-hand picture, and no underlying representation of the form "L1 parallel L2 transversed by L3..." The second axiom asserts in pictures the congruence of opposite angles of intersecting lines. The goal of the work here is to describe such formal systems, wherein picture axioms support formal deductions as rigorously as algebraic ones shown in Figure 1(a). Though the geometry rules of Figure 1(b) suggest what it might be like to do imaginal deductive inference, the example is very incomplete. In particular, the deduction requires that Axiom 2 be applied to the picture on the right-hand side of Axiom 1. No machinery has been defined that lets Axiom 2, which has only two lines in its picture, apply to the right-hand side of Axiom 1, where there are three lines. Somehow the left-hand side of Axiom 2 must stand for more than what meets the eye, more than the exact two-line picture on the paper. In conventional logics, like those illustrated in Figure 1(a), the corresponding problem is handled by the well-articulated mechanisms of variables, quantification,

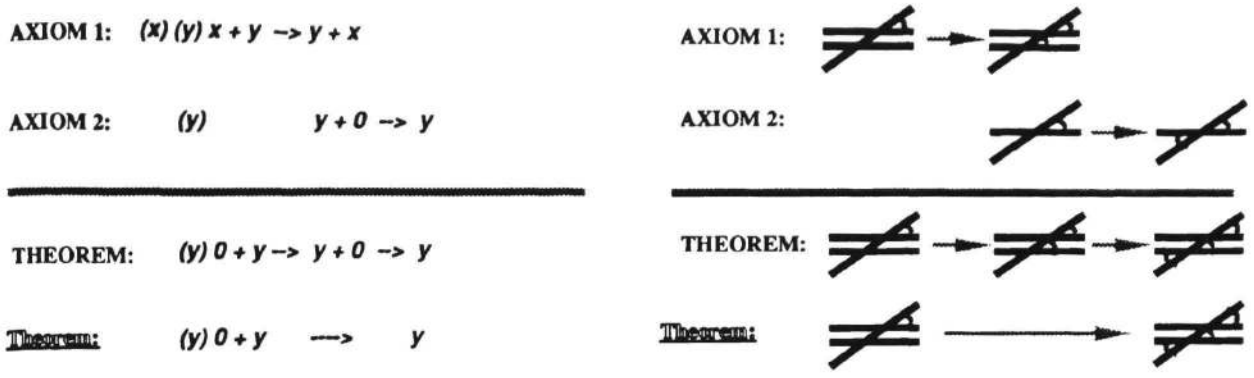


Figure 1. (a) A deductive system for algebra manipulation, showing the chaining of axioms for commutativity and zero-*right*-additive identity to get zero-*left*-additive identity. (b) A structurally analogous deduction in a hypothetical picture system to do Euclidean geometry proofs. In each case the first version of the resulting theorem preserves the intermediate link in the deductive chain, the second one omits it.

unification, bindings, etc. But where are the variables and quantifiers in the graphical axioms of Figure 1(b)? How can something like unification be accomplished? Since these mechanism have no direct analogs in the imaginal domain, it has been necessary to analyze what these mechanisms accomplish, to produce a list of generalized properties needed in a deductive formal system. These desired properties can

2. *Introduction to a General Theory of Deduction*

The difficulty for pictorial deduction in Figure 1(b) arises from the need to take the pictures in Axiom 2 other than literally. Consider the alternative, illustrated with the "Happy Face" system of Figure 2. Assume that *exactly* the picture appearing as the left-hand side [LHS] of Axiom 1 yields *exactly* the picture on the right-hand side [RHS] of Axiom 1, and *exactly* the picture appearing as the LHS of Axiom 2 yields *exactly* the picture on the RHS of Axiom 2. From these two axioms, one can conclude the "Happy Face Theorem", that *exactly* the LHS picture of Axiom 1 yields *exactly* the RHS picture of Axiom 2. Given a different frowning face, one of a different size or shape, even a single deductive step would be impossible, to say nothing of a chain of two.

2.1 The generative power of high-level rules

The literal deduction of the Happy Face Theorem is perfectly rigorous; it is also very impoverished. Many literal (*low-level*) rules would be needed to even approach some interesting behavior. What is required is a way to write *high-level rules*, i.e., a notational system whereby a whole set of low-level rules are specified at once. Such generative power is one of the important accomplishments of variables and quantifiers in standard sentential formalisms. Thus, the algebraic rule, $\forall x \forall y \;\; x + y \;\rightarrow\; y + x$, can really be considered as the name for a set of low-level arithmetic rules, $\{1 + 3 \rightarrow 3 + 1,\; 2 + 7 \rightarrow 7 + 2,\; 0 + 4 \rightarrow 4 + 0,\; \cdots\}$. Familiar machinery here generates the set of instance rules from the high-level algebraic expression using the Cartesian product of the instantiations of the universally quantified variables. For pictures, where it is not at all clear where or how to put in variables, a more general understanding of high-level rules is needed.

Low-level rules of a system associate ordered pairs ($a \rightarrow b$, $c \rightarrow d$, $e \rightarrow f$, $b \rightarrow f$, *etc.*) of "low-level" objects, for example, pairs of individual expressions of arithmetic or pairs of specific pictures of faces. The basic deductive inference is of the form, "given a and $a \rightarrow b$, conclude b," and enables deductive chaining of the low level rules, like,

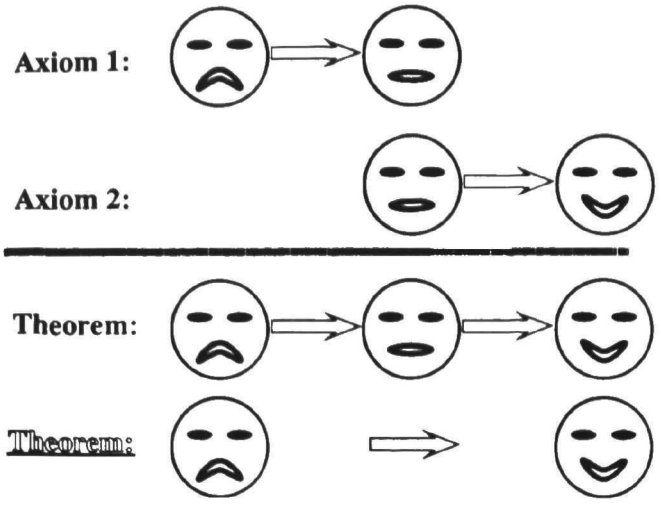


Figure 2. The "Happy Face Theorem". If the rules here are taken literally, i.e., as applying to only exact matches to the faces shown, a rigorous but impoverished deductive system results.

$$Given \quad a \rightarrow b \quad and \quad b \rightarrow f, \quad conclude \quad a \rightarrow f.$$

The happy-face deduction of Figure 2 , is an example of such a low-level chain.

A *high-level* rule specifies a whole set of such low level rules at once. This set of low-level associations defines a mapping between two sets of low-level objects. So a high-level rule is fundamentally a mapping,

$$f: A \rightarrow B, \quad i.e., \quad a_1 \rightarrow b_1, \; a_2 \rightarrow b_2, \; \cdots$$

between two sets (its Left- and Right-Hand sets), and as such may admit specification mechanisms other variables and quantifiers. Thus for a deductive system in a given domain of low-level objects (e.g., pictures), one needs devices for

(P1) the specification of sets of low-level objects, and
(P2) the specification of mappings between those sets,

that are suitable for the objects in the domain, and computationally tractable. Suitability, for example, is at issue for the straightforward application of the device of variables and quantifiers for the specification of sets of pictures. Computational tractability is required for the operations implicit in the application of the function, f , to a candidate object x. In particular, both the applicability of the function (the test of whether $x \in A$) and the actual result of the function must be computable with reasonable resources.

2.2 The deductive power of high-level rules

The generative power of high-level rules, i.e., their ability to specify a whole set of low-level rules at once, is only one important desideratum of a deductive system; a second is the ability to do high-level deductive chaining. A high-level deductive chain takes two high-level rules f and g and derives a third high-level rule h, subject to the condition that the low-level rules in h are all implied by those of f and g. That is, whole sets of low-level rules are deductively chained by chaining high-level rules. In this way, high level axioms are chained, as shown in Figure 1, to derive new "theorems". In terms of treating high-level rules as mappings, one wants to take two mappings, $f: A \rightarrow B$ and $g: C \rightarrow D$ and say as much as possible about applying one to the output of the other, i.e., forming a function composition. If the sets B and C are equal, the composition is straight forward,

$$Given \quad f: A \rightarrow B \quad and \quad g: C \rightarrow D \quad and \quad B = C, \quad conclude \quad g \circ f: A \rightarrow (B = C) \rightarrow D.$$

However if $B \neq C$ then the maximal composition is mediated by the intersection, $B \cap C$. Writing a bit

loosely (i.e., applying f and g to sets in the obvious way), we have

Given $f: A \rightarrow B$ *and* $g: C \rightarrow D$, *conclude* $f^{-1}(B \cap C) \rightarrow (B \cap C) \rightarrow g(B \cap C)$.

The computation of maximal partial composition may be thought of as having three critical steps. First is the computation of the intersection $B \cap C$. In standard deduction, this is accomplished by *unification* and yields information on variable bindings (e.g., y is bound to 0 in the algebra example of Figure 1(a)). The second and third steps map this intersection backwards by f^{-1} into A and forwards by g into D. In standard deduction, this is accomplished by substituting the variable bindings resulting from the unification, into the LHS of the first rule and the RHS of the second rule. Thus in any system where (P1) and (P2) above were defined, any mechanism for

(P3) the computation of $B \cap C$, *and*
(P4) the computation of $f^{-1}(B \cap C)$ *and* $g(B \cap C)$,
i.e., a way to test whether $a_i \in f^{-1}(B \cap C)$, *and if so to find* $d_i \in g(B \cap C)$,

would then enable high-level deductive chaining.

While the ideal is to have a system that supports (P3), one could settle for less. For example, if it were only possible to compute subsets of $B \cap C$, one could still make valid, though not maximal, high-level deductions. Note also that systems with no high-level deductive chaining could still be useful. For example, most current expert systems never prove theorems. They use only the generative power of high-level rules, taking some particular low-level instance from the world and cranking it through a rich system of high-level rules to deduce new low-level conclusions. Thus it is quite possible that a picture system that only satisfied (P1) and (P2) could still be quite important.

2.3 Candidate mechanisms for (P1)-(P4)

There are many strategies that can be used to satisfy various of the desiderata, particularly (P1) and (P2) (specifying sets of pictures and picture-to-picture mappings). A few of the more promising general approaches, fairly different in character from the use of quantified expressions of variables, are sketched below. Versions of the first two of these have been used in the the BITPICT system (see below), a version of the third in another system we are working on.

2.3.1 Instance-plus-transformation strategies. A set can be specified by a canonical instance a_0 and a set $T=\{t_i\}$ of permissible transformations of that instance. A rule can similarly be defined by giving a canonical low-level rule, $a_0 \rightarrow b_0$, and a set of transformations, T. The corresponding high-level rule is taken to map $t_i(a_0) \rightarrow t_i(b_0)$. For example, if T is the set of size rescaling transformations, and $a_0 \rightarrow b_0$ is the low-level, "sad-face" rule (Axiom 1) of Figure 2, then the result is a high level rule which maps the sad face at size i to the neutral face resized to size i. Similarly taking T to be the set of affine transformations of the plane[1] and the axioms of the Euclidean Geometry example as canonical rules, would yield a rigorous high-level picture deduction system (though still providing only part of the needed generality).

2.3.2 Feature-set strategies. One class of systems that fulfill all of (P1)-(P4) quite naturally arises if low-level objects can be characterized by sets of features. Then subsets of features can be used to specify sets of objects possessing the given features. There is a natural partial ordering of both object and feature sets by containment, forming a dual lattice structure upon which the intersections required for unification can be easily calculated, and for which there are natural notions of generalized binding.

1. Note that affine transformations are typically discussed in conventional sentential algebraic terms (e.g., by certain matrix multiplications), but that is not inherent. They can be represented and implemented otherwise (e.g., optically).

2.3.3 Grammar-based strategies. Another way to specify a set is as the language of a grammar. Thus, for example, most standard logics have a context-free string grammar that specifies the well-formed formulas of the formalism. The axioms map between certain specified subsets of the well-formed formulas. In particular, the left-hand side of many rules can be thought of as generated by a sub-grammar (e.g., "X + Y" as "expression1 : plus-operator : expression2"). A parser builds a parse-tree over the input to the left-hand side of the rule, and gets the object on the right side of the rule by manipulating the parse tree (e.g., flipping the two descendants of the "+" operator, to get "Y + X").

Such an exercise need not be restricted to string systems. There are various grammars for spatially structured objects, for example array and matrix grammars [4] [5] or picture grammars[6]. Collaborations with Karen Lochbaum led towards a system that specifies sets of simple pictures for the LHS of rules using context-free array grammars,with high-level rules operating by transformig the resulting spatially structured parse-trees.

3. The BITPICT deduction system

Several of the strategies just outlined have been combined in a particular system, the BITPICT system, and a prototype virtual machine now exists, running on a Symbolics Lisp Machine. It is a particularly simple formal system which, though not powerful enough to handle the Euclidean geometry example in full generality, is sufficient to give a better sense of what picture deduction systems might be like.

The architecture of the BITPICT system is similar to that of standard production systems. That is, like classical production systems [7] [8] [9], it has a set of rules that interact via a shared blackboard. Each rule has a left-hand side that constantly looks at the blackboard for a match. When a rule finds a match, its right-hand side directs how to change the contents of the blackboard. The change is made, and the process iterates. The rules are engineered so the evolution of the blackboard solves the intended problem.

Unlike standard production systems, where the structures on the board and in the rules are typically strings of symbols, in the BITPICT system they are simple picture fragments. Picture fragments are matched and replaced and problems are solved by the fragment-by-fragment evolution of the picture on the blackboard.

The universe of pictures considered by the BITPICT system are bitmaps, i.e., regular grids of picture elements (pixels) that are either black or white. A *bitpict* is a small such bitmap, and is meant to be the specifier for a whole set of other bitmaps, in this way performing a function analogous to quantified algebraic expressions specifying sets of arithmetic ones. A bitpict specifies the set of all bitmaps which contain that bitmap fragment somewhere. A *bitpict rule* is an ordered pair, LHS->RHS, of conformal bitpicts, and is taken to specify a mapping between the two corresponding sets of pictures. The mapping simply associates left- and right-hand pictures that differ only in that the LH bitpict has been replaced by the RH one. (See Figure 3.) There are two components to how bitpicts specify sets of pictures: (1) the use of a piece of a picture to specify the set of pictures containing that piece and (2) translation invariance, i.e., fact that the piece may occur at any vertical or horizontal position. The first is an example of a feature-set strategy: a pixel being black or white is a feature, and picture sets are specified by a spatially defined collection of such features. As mentioned earlier, notions of unification fall out easily from the resulting lattice. Let A and B be two bitpicts (but with translation disallowed, i.e. each at fixed positions). The *intersection* of their corresponding picture sets is a set whose specification is the pixel-wise *union* of A and B (or nil if they disagree at points of overlap). The second aspect, translation invariance, is an example of the instance-plus-transformation strategy: the i,j-th translation of the left-hand bitpict is mapped into the corresponding i,j-th translation of the right-hand bitpict. In fact, the BITPICT system can variously allow translations, rotations (0, 90, 180, or 270 degrees), and reflections. (The small crossed vertical and horizontal lines appearing like "+" sign, under the circle bitpict is an icon indicating that this particular rule holds for vertical and horizontal translations only.)

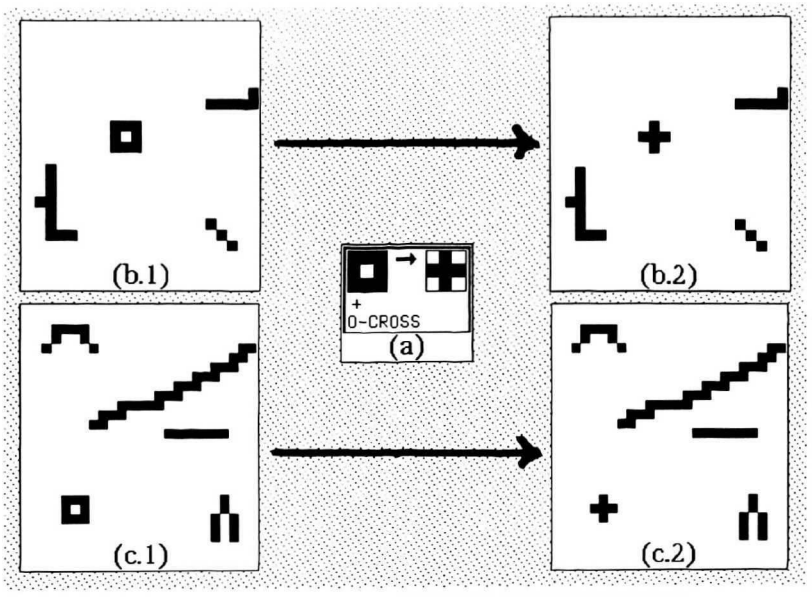


Figure 3. (a) A BITPICT rule mapping the 3x3 grid of pixels containing a circle to the 3x3 grid containing a cross. (b.1) and (c.1) show pictures in the left-hand set of the rule. (b.2) and (c.2) show the corresponding pictures in the right-hand set.

3.1 A BITPICT example: Counting the tangled forest

An example of the the BITPICT system solving a spatial problem with graphical deductions is presented in Figure 4. The problem, counting the disconnected components in a tangled forest of bifurcating trees, is not easy to represent in a sentential way, yet has a very natural solution with bitpicts.

Part (a) of the figure shows a sample tangled forest. The first set of rules, (b), simplify the problem, basically by nibbling back the tips of the branches, nibbling at straight, corner and "T" sections respectively. The rules neither create nor destroy connected components, and so each component is gradually reduced, (b.1), to a dot (b.2).

The next rules, (c), move the dots down, and when they hit the bottom of the window, to the right, as shown in (c.1), until they are all canonically aligned in the lower right corner, (c.2). Again the invariant of number of components is preserved, and the problem of counting trees has been reduced to counting these neatly arrayed dots.

The final rules, (d) count the aligned dots by converting each to a vertical bar (an "I") and from there to Roman numerals, with rules for simplifying (e.g., five I's to a V), maintaining alignment (e.g., shifting V's and X's right as needed), and sorting (e.g., VX to XV). The final result is the Roman numeral, XII, indicating the number of trees in the original tangled forest.

The critical computational point is that the evolving state of the blackboard is governed by the picture-to-picture mapping rules, with no need for underlying sentential representation.

4. Discussion

In addition to solving certain spatial problems and implementing graphical symbol manipulation algorithms (e.g., Roman numeral counting and arithmetic), the BITPICT system is also useful for auto-animation (e.g., cartoon figures that move themselves according to rules), for certain kinds of naive physics models (e.g.,Figure 5), and for the analysis of graphical computer interfaces.

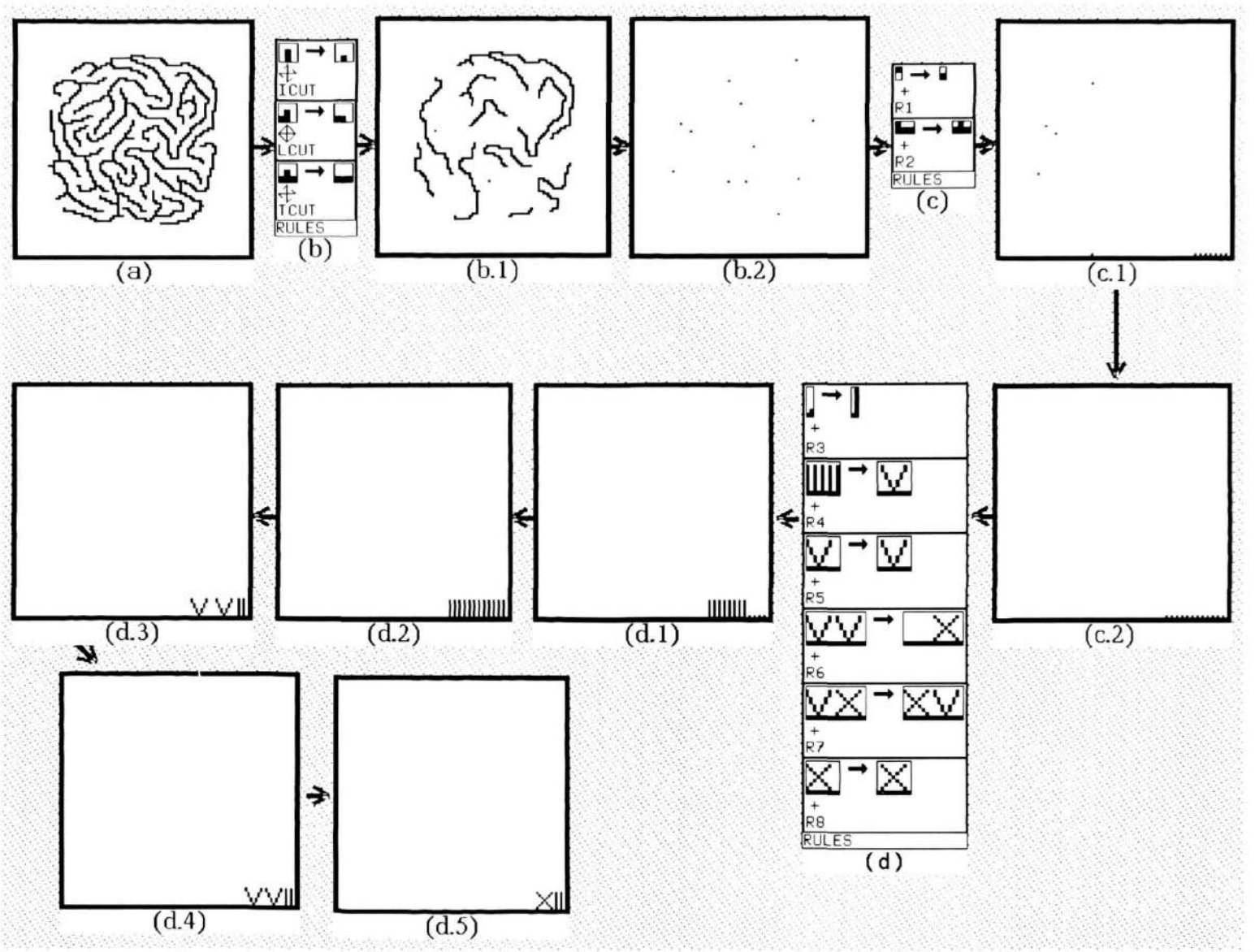


Figure 4. Counting the tangled forest. (a) A tangled forest of bifurcating trees. The rules, (b), nibble at the tips of straight, corner "L", and "T" sections, reducing each component, (b.1), to a dot (b.2). The rules, (c), move the dots down, and to the right, (c.1), into the lower right corner, (c.2). The rules, (d), count the aligned dots in Roman numerals, yielding the answer, XII.

The BITPICT system has a number of limitations, arising from its tie to the rectangular pixel grid, its current 2-D character, and the flatness of its representations (without the structural depth of grammar approaches), but it has been important in showing what imaginal deduction might be like.

REFERENCES

1. Shepard, R.N., and Metzler, J., Mental rotation of three-dimensional objects, *Science*, **171**, 1971, 701-703.
2. Kosslyn, S.M., *Image and Mind*, Cambridge, MA: Harvard University Press, 1980.
3. Lindsay, R.K., "Images and inference" *Cognition*, **29**, 1988, 229-250.
4. Siromoney, G., Siromoney,R. and Krithivasan, K., Picture languages with array rewriting rules, *Information and Control*, **22**, 1973, 447-470.
5. Rosenfeld, A., Array and web grammars: an overview. In A. Lindenmayer and G. Rozenberg (Eds.), *Automata, Languages, Development*, North-Holland, 1976, 517-529.
6. Lakin, F., Spatial Parsing for Visual Languages. In S.K.Chang, T.Ichikawa, and P.Ligomenides (Eds.), *Visual Languages*, New York: Plenum, 1986,35-85.
7. Post, E.L., Formal reductions of the general combinatorial decision problem, *American Journal of Mathematics*, **65**, 1943,197-268.
8. Newell, A., and Simon, H., *Human Problem Solving*, Englewood Cliffs, NJ: Prentice-Hall, 1972.
9. Waterman, D.A. and Hayes-Roth, F., *Pattern-Directed Inference Systems*, NewYork: Academic Press, 1978.

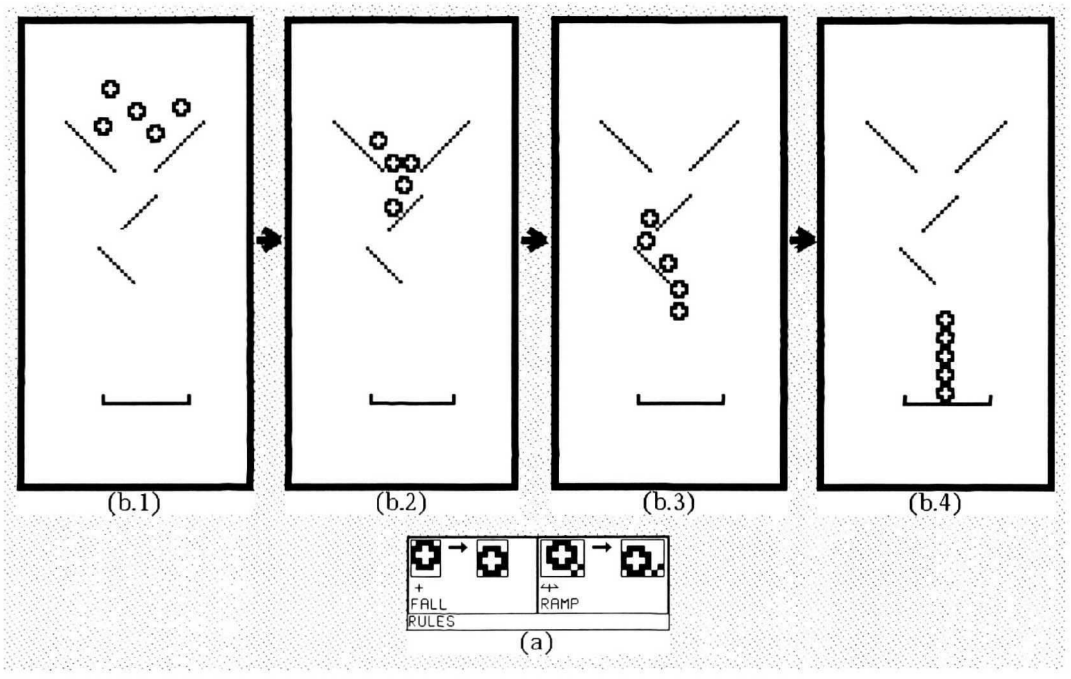


Figure 5. Balls and ramps - a naive physics model. (a) Rules for straight fall (translation invariant) and rolling down a piece of ramp (translation and reflection invariant). (b.1) An initial configuration. (b.2),(b.3) Samples from intermediate states in the deduction. (b.4) Final configuration.

Integrating Imagery and Visual Representations *

B. CHANDRASEKARAN AND N. H. NARAYANAN
Laboratory for Artificial Intelligence Research
Department of Computer and Information Science
The Ohio State University
Columbus, OH 43210

The issue of propositional versus analogic representations for visual information has been debated extensively in cognitive psychology. In this paper we argue that issues arising from this debate can be effectively addressed by postulating a common mechanism underlying visual processing and mental imagery which, while representing information symbolically, can manipulate visual aspects of perceived objects in imaginal forms. The central components of this mechanism are representations resulting from visual perception, called visual representations, and a special-purpose architecture specific to the visual modality. The experience of mental imagery arises from the interpretation of visual representations by the modality-specific architecture. This architecture also allows operations specific to the visual modality to be performed on visual representations. Thus, the modality-specific architecture is the key that integrates the experience of analogic imagery and symbolic visual representations. We argue that there is no real opposition between a belief in the need for some form of analogic phenomena to explain imagery and a belief in the realization of such phenomena from symbolic structures, and that a middle course can be charted between the purely analogic and the purely propositional views while retaining the advantages of both.

1. INTRODUCTION

The representation of visual information and its relation to the phenomenon of mental imagery have attracted a great deal of attention from cognitive psychologists. There has been considerable debate (Anderson, 1978; Kosslyn & Pomerantz, 1977; Kosslyn, 1981; Pylyshyn, 1981) about postulating separate analogic representations for imagery as opposed to uniform propositional representations. The representation of visual information must be "picture-like", in some sense of the term, to one camp because they see evidence that some aspects of human reasoning are better explained by exploiting special properties of this modality. Scanning, relative distance estimation, and direction finding are examples of mental actions, explanations of which benefit from postulating a pictorial representation. For researchers subscribing to this view, what is important is that any ultimate proposal for mental representations should be able to explain this differential use of modality-specific operations. On the other side there are people who view propositions as the basic currency of mental representations. They believe that the use of propositional representations unite conceptual and perceptual processes. They view the advocacy of image-like representations as a rejection of their claims about the generality of propositional representations. They argue that the use of visual operations can be explained by propositional representations and processes as well. Many such arguments are given in (Anderson, 1978; Pylyshyn, 1981).

This is the dichotomy between the positions taken by proponents of analogic and propositional representations. But are these two views really mutually exclusive contenders for the position of the "ultimate" theory of visual representations? Maybe what is missing is the realization that a mechanism that preserves the significant properties of both analogic and propositional representations can account for the supportive experimental evidence presented by researchers on both sides of the issue. This is the insight that led to the ideas presented in the rest of this paper.

---

* Ideas in this paper have benefited from our discussions with Stephen Kosslyn. This research has been supported by DARPA & AFOSR contract # F-49620-89-C-0110 and by AFOSR grant # 890250. First author's email address is chandra@cis.ohio-state.edu.

2. TYPES OF VISUAL REPRESENTATIONS

In the literature, representations that result from actually perceiving objects are called perceptual representations and representations that underlie mental imagery are called imaginal representations. We refer to these collectively as visual representations in this paper. In this section three types of visual representations are described.

The analogic-propositional debate centered around two kinds of representations. One is called an analogic representation and the motivation for it stemmed from the phenomenon of mental imagery. While few deny the existence of mental imagery, the hypothesis that analogic representations underlie mental imagery has been questioned. Those who subscribe to this hypothesis, whom we call analogists, are impressed by what they see as evidence for subjects' preferential use of pictorial operations (e.g: scanning) on mental imagery and to account for this they ascribe to this representation certain properties that are usually associated with pictures. Such operations have a special status in analog representational theories, but not in propositional theories. Kosslyn (1981) presents a concrete explication of an analogic theory of visual representation. He proposes two levels of representations: one "deep" representation that is abstract and not experienced directly and a "surface" representation that supports mental imagery. A surface representation is pictorial in nature since it depicts an object by regions of activation in the visual buffer which is an analog medium. This two-level view of imagery (i.e., surface displays generated by comparison and/or transformation processes from a deep structure) is used in (Kosslyn & Schwartz, 1977; Kosslyn & Pomerantz, 1977; Pinker, 1980) as well.

The other type of representation is called propositional (and its advocates, propositionalists). It is made up of propositions. A proposition consists of symbols, but it is more than a mere collection of symbols. Propositions have identifiable predicate and argument constituents, bear truth values, and have rules of formation (Anderson, 1978). Thus a proposition has both a fixed syntactic form and a fixed semantic content. In the propositional view of mental representations, propositions encode knowledge about objects in a perceived scene and require interpretation for their semantic contents to be accessed by processes operating on them. Propositional theories have an underlying implication that the rules which govern how propositions are interpreted are independent of the perceptual or cognitive modality that produced the propositions. In other words, a general purpose mechanism that is not modality-specific is assumed to operate on propositional representations. For example, according to propositionalists, the propositions (*left-of A B*), (*smell-of C pungent*) and (*rate-of inflation high*) will be handled by mental processes in a uniform manner that does not reflect the distinction that one describes a visual attribute, another an olfactory attribute, and the third is a conceptual assertion. Thus the intrinsic property of propositional representations is their uniformity of representation and processing across modalities or "faculties".

There is a third type of representation, called a discrete symbolic representation. A discrete symbolic representation comprises structures of discrete or atomic symbols composed according to well-defined rules of formation. A symbol is just a token and, by itself, is devoid of any meaning. Its semantics derives from the *architecture* of the system it resides in and from *how* it is used by processes operating on it. Therefore it is conceivable that the same symbol may be interpreted differently by different architectures and processes. This is an important distinction. Consider the symbol "car"[1] appearing as the first element of a list. In an architecture designed specifically to execute Lisp programs, this symbol will trigger a process that extracts the first element of a list. In a different architecture the same symbol may have a different meaning. The meaning of a program (a program is nothing but a discrete symbol structure) is given by its operational semantics which specifies the operations performed and their effects on the inputs (which are also discrete symbol structures) and the compiler enforces this semantics. We propose

[1] "Car" is a Lisp command that extracts the first element of its argument, which should be a list.

that the discrete symbolic representation in the general sense is also a contender for the mental representation of visual information. The operational semantics of such a visual representation specifies the operations allowed by the visual modality-specific architecture and their effects on the representation and the architecture enforces this semantics.

Both propositional and discrete symbolic representations are made up of symbols and composed according to specific rules of formation. However, unlike propositional representations, a discrete symbolic representation does not necessarily need to have a truth value or have predicates and arguments as constituents. In other words, a proposition is a special type of discrete symbolic structure whose operational semantics is truth-preserving[2]. Thus discrete symbolic representations are more general than propositional representations. The only commitment entailed by the discrete symbolic representation is that it is composed, in a principled manner, of discrete symbols and interpreted consistently by the underlying architecture and processes operating on the representation. While the discrete symbolic representation has a fixed syntactic form, its semantic content is defined relative to the underlying architecture and the processes that operate on it. Properties exhibited by such a representation are not intrinsic to the representation itself, but stem from mechanisms (which may well be sensory-modality specific; this is one of the claims made later in this paper) that support and operate on the representation.

3. THE IMAGE REPRESENTATIONAL SYSTEM

We referred before to a debate in psychology between analogists and propositionalists about the nature of visual representations. The debate is often stated in terms of several "versus" formulations:

1. Analog versus symbolic: Some picture-like analog representations versus discrete symbolic representations.
2. Analog versus propositional: Picture-like representations versus propositional representations.
3. Sensory modality-specific representations and interpreters versus sensory modality-independent representations and a uniform interpreter.

In the literature on this debate, discrete symbolic representations are equated with propositional representations, i.e., 1 and 2 above are generally supposed to make the same distinctions. However, as we just discussed, propositional representations are a special form of discrete symbolic representations. Additionally, because of the uniformity of interpretation that applies to propositional representations, propositionalists are necessarily led to a sensory modality-independent view of representation. To the extent that analogic representations involve interpretations that give a privileged status to some operations over others depending on the sensory modality, there is a genuine opposition between analogists and propositionalists.

The burden of this paper is that, on the contrary, there is no real opposition between a belief in the need for some form of analogic representations and a belief in the realization of such a representation from symbolic structures. The facile equation of propositional representations and symbolic representations has, in our view, set up a false opposition in 1 above. The root of this misconception arises from the assumption that symbolic representations necessarily need to run on general purpose computers. We argue that in fact one can have analogic, i.e., visual modality-specific, representations which are also in principle symbol structures. The key to this possibility is that representations for different sensory modalities are run on interpreters which provide privileged operations that are specific to that modality. Of course, these representations, to the extent possible and relevant, need to be coordinated with corresponding representations in

---

2 Note that logic has two distinct uses. One is as a representational language and the other as a metalanguage in which a representational theory can be described and analyzed. Logic as a programming language is an example of the former and logic specification of program semantics is an example of the latter. Here we are concerned only with the former use of logic in propositional representations.

other sensory modalities as well as the general cognitive architecture.

We propose that properties exhibited by visual representations stem from an underlying mechanism, which we call the Image Representational System (IRS). The IRS contains a special purpose architecture that is specific to the visual modality. Visual representations reside in this architecture and the architecture interprets the representations in a way that allows visual operations (e.g., scanning, relative position estimation etc.) to be performed on them. It is this interpretation that gives rise to mental imagery. Thus the analog nature of mental imagery arises from the interaction between visual representations and the visual modality-specific architecture.

The central components of the IRS are the visual representations that result from perception, called Image Symbol Structures (ISS), the underlying specialized (to the visual modality) architecture in which these structures reside, the interpretation of ISS (which gives rise to mental imagery) that the architecture produces, and the visual operations that it provides. An ISS is a hierarchical discrete symbolic[3] representation, similar in spirit to the 3-D sketch of Marr and Nishihara (1978). It is the end product of visual processing that starts from the retinal image.

The ISS has both syntactic form and semantic content. When an ISS is interpreted by the architecture, its semantic content can be experienced as mental imagery. These interpretations as well as the ISS itself function as inputs to high level visual processes for object recognition etc. When an object represented in the ISS is recognized and labelled, that facilitates the evocation of non-visual (conceptual) knowledge about it. The visual modality-specific architecture of the IRS also provides basic operations such as scanning on the ISS. These have concomitant effects on its interpretation as well. Visual processes can manipulate the Image Symbol Structures by invoking these operations. Fig. 1 illustrates[4] this role of the IRS. A mechanism like the IRS can account for the preferential use of visual operations on mental images based on underlying non-analogic representations.

4. THE IMAGE SYMBOL STRUCTURE

Visual perception, according to the theory of Marr and Nishihara (1978), consists of the transformation of the primal sketch obtained from the retinal image into a $2^1/_2$-D sketch and then into a 3-D sketch which feeds into shape and object recognition processes. The most interesting aspect of this theory is its use of parametrized volumetric primitives in the 3-D sketch. Our conception of the Image Symbol Structure has been inspired by Marr's theory. The ISS is defined to be the internal representation of a perceived real world scene. It is a hierarchical compound structure made up of primitives.

Primitives of shape, texture, color and other visually perceivable attributes are assumed to be available to the IRS. Each primitive is parametrized. For example, a shape primitive may have its relevant dimensions as parameters. These parameters are not absolute measurements, but have relative meaning within an internal reference frame. That is, they serve as yardsticks that facilitate attribute value comparisons with other primitives of the same type.

The ISS is structured as a hierarchy of descriptions with levels that decrease in grain size,

---

[3] Quite independent of the debate about visual representations being analogic or propositional, there is another ongoing debate about mental representations in general being symbolic versus connectionist. In this paper we argue that visual representations can be discrete symbolic while preserving analogic properties. This argument can be extended to the connectionist position as well. The aspects relevant to our proposal are the informational content of the representations (Chandrasekaran, Goel, & Allemang, 1988), how this visual information is organized (e.g., structured as hierarchical composites built up from primitives of visual attributes, see the following section), and how the representations get interpreted by the architecture.

[4] A similar depiction was originally suggested to us by Bruce Flinchbaugh.

or alternately, increase in resolution. The description at each level is made up of appropriately parametrized primitives corresponding to objects delineative at that level's resolution. The topmost level describes the image coarsely while the lowest level describes it in terms of the finest details captured during perception. At each intermediate level more details get added to the descriptions of image components from the level above. The ISS encodes only the intrinsically visual (and therefore internally visualizable) aspects of a scene. Non-visualizable aspects (for example, knowledge about the weight of an object) are part of the conceptual knowledge associated with the scene and not part of the ISS. Also, the ISS is neutral with respect to recognition; it represents objects in terms of visual attributes, but does not "name" or label the objects that it represents.

We term those parts of an ISS that together correspond to an object or a delineative part of an object, an S-percept ("symbolic percept"). An ISS is thus made up of multiple S-percepts and an S-percept may itself be composed of other S-percepts. It is essentially a description that contains (only) visual aspects of a delineative object or its part, in terms of parametrized primitives. For example the S-percept corresponding to an apple will consist of primitives that describe its shape, color, shiny texture etc., but not its taste or nutritional value. In addition to parametrized primitives an S-percept also contains descriptions of spatial relations among the primitives.

Each S-percept has a corresponding mental image that results from its interpretation. This entity is called an A-percept ("analogic percept"). An A-percept exhibits all visual attributes and spatial orientations described by the corresponding S-percept. For instance, an A-percept corresponding to an S-percept of the form {cylinder (diameter:2, length:4), color(red)} will be the mental image of a red cylinder of diameter 2 units and length 4 units. Thus the S-percept of an object is a symbolic entity whereas the A-percept of the object is an imaginal entity, and the two may be thought of as two sides of the same coin.

Thus the ISS may be viewed as an internal representation of a perceived scene that functions as a symbolic description as well as an algorithm for the composition of a mental image by the visual modality-specific architecture. The following analogy should explicate how ISS, S-percepts, and A-percepts are related. Consider a robot standing beside bins containing cubes, cylinders and other geometric objects of various dimensions and colors. Assume that it is possible to write a description of any structure made up of these geometric objects in an abstract language that the robot can interpret. Upon loading such a description into the robot it is capable of picking up objects of appropriate shape, size and color and building that structure. Conversely, if a structure made up of the geometric objects is provided, the robot can produce a description of that structure in the abstract language. Assume that if some component in this structure (or the abstract description) is removed or replaced with another, it is possible for the robot to sense the change and correspondingly change the abstract description (or the structure). This situation is similar to the mechanism comprising IRS and ISS. The robot is analogous to the special architecture of the IRS, the abstract description is analogous to an ISS, parts of the abstract description are analogous to S-percepts, the geometric objects are analogous to A-percepts, and the structure built by the robot is analogous to the interpretation of an ISS.

Operations performed on S-percepts (A-percepts) have concomitant effects on A-percepts (S-percepts). The mental image of a scene results from accessing the ISS corresponding to that scene and bringing into the IRS a description of the scene from the appropriate level in the ISS hierarchy. Some examples of basic operations performed on mental images are changing the relative position of an object, enlarging or zooming in on an object, and scanning. In the IRS these are not analogic operations. Rather, changing the position of an object can be achieved by appropriately modifying spatial relations among S-percepts in the ISS. Zooming in on an object corresponds to selectively bringing in a more detailed (lower level) description of the corresponding S-percept from the ISS hierarchy. Scanning is the process of moving one's fixation point from object to object in the mental image and bringing in more material (from the

ISS) as the fixation point approaches the mental horizon (Pinker, 1980).

Fig. 2 shows an example that illustrates the inherent correspondence between S-percepts and A-percepts. It is, however, meant only as an analogy. Consider a two-dimensional array of two-valued logic elements with light bulbs attached to each element so that an element has value 1 if and only if its bulb is on and an element has value 0 if and only if its bulb is off. A pattern stored in this array may then be viewed either as an image or as a set of multi-dimensional bit vectors. Now interface this array with symbolic processes through a "vector set<->symbol" convertor that can translate between parametrized two-dimensional shape descriptions like "rectangle(2,3)" - meaning a rectangle of breadth 2 units and length 3 units - and bit vector sets. Similarly interface the array through a "photoreceptor-actuator" matrix with pattern-manipulating processes. This matrix is capable of sensing images displayed on the array as well as actuating bulbs in the array in response to input from the pattern-manipulating processes. Then the symbolic (pattern-manipulating) processes can input a symbolic description (image) to the vector set<->symbol convertor (photoreceptor-actuator matrix) and a corresponding bit vector set will be loaded onto the array which can then be sensed by the photoreceptor-actuator matrix (vector set<->symbol convertor) and the resulting image (symbolic description) can be sensed and modified by the pattern-manipulating (symbolic) processes. The modified image (symbolic description) can then be fed back into the symbolic (pattern-manipulating) processes in a similar fashion. This is an example of a mechanism that consists of a representation (bit vector set) residing in a specialized architecture (consisting of the array, photoreceptor-actuator matrix and the vector set<->symbol convertor) which generates symbolic descriptions and image-like interpretations of the represented entity and also acts as a two-way channel between symbolic and pattern-manipulating processes. In this example the symbolic shape descriptions are analogous to S-percepts and the patterns depicted on the array are analogous to A-percepts. The bit vectors are analogous to the primitives that S-percepts are composed of.

5. RELATED WORK AND DISCUSSION

Work by Kosslyn and Schwartz (1977) on two dimensional mental images and by Pinker (1980) on 3-D images report similar models. Kosslyn (1981) provides a cognitive theory that utilizes two distinct representations - deep non-pictorial representations and surface representations (patterns) in a visual buffer. There are similarities and differences between this and our model. The ISS and A-percepts may be viewed as deep and surface representations respectively. The IRS can have a component similar to the visual buffer as a medium for A-percepts. However, the ISS is a hierarchical multi-resolution representation which consists of object descriptions composed of visual primitives, as opposed to propositional encodings in the form of lists and literal encodings. Also, an architecture specialized for visual perception is central to our theory.

Pylyshyn (1981; 1984) has also made significant contributions to the imagery debate. He uses the concepts of tacit knowledge and cognitive penetrability to argue against a purely "analogue" position. The IRS is not in opposition to these concepts. In fact, some cognitive processes that operate on ISS or A-percepts may be (and some may not be) cognitively penetrable and thus influenced by tacit knowledge, overt instructions etc. However, we postulate that the interpretation of the ISS by the visual modality-specific architecture and the basic operations that the architecture provides on ISS and A-percepts are not cognitively penetrable and that these are in fact properties of the functional architecture (as defined by Pylyshyn) of visual perception and mental imagery.

The IRS does not provide for a separate analogic representation. However, the preferential treatment of modality-specific operations that analogists require is preserved in the IRS. On the other hand, while the IRS does not confirm to propositionalists' belief in a uniform mechanism for all of cognitive and perceptual processing and instead advocates modality-specific architectures, propositions are in fact one type of discrete symbolic representation and thus

advantages of a uniform representation that a propositional theory provides are preserved in the IRS as well.

Our proposal is also related to the question about how a purely syntactic system, such as a Turing Machine, can make connections to semantics, except in an arbitrary way. For example, this question lies at the heart of Searle's Chinese Room argument which seeks to show that computer programs cannot understand the meanings of symbols they manipulate. The IRS shows how modality-specific architectures preserve some aspects of the semantics of the world in such a way that the symbols can be seen to be "grounded" in perception as Harnad (1988) points out. This issue is described in more detail in (Chandrasekaran & Narayanan, in press).

6. CONCLUSION

In this paper we provide a description of how visual knowledge can be represented within the computational framework of discrete symbolic representations in such a way that both mental images and symbolic thought processes can be explained. Thus we answer the question "how can a percept that appears in the mind's eye as an image be symbolic?" by saying that it can indeed be so, given that a special purpose architecture, providing privileged visual operations, underlies visual representations. We propose a mechanism called an Image Representational System that provides interpretations of and visual modality-specific operations on symbolic visual representations, called Image Symbol Structures. An Image Symbol Structure is a hierarchical multi-resolution structure composed of S-percepts. S-percepts are made up of parametrized symbolic primitives of visual attributes such as texture and color. S-percepts represent delineative objects or parts of objects seen. A-percepts are interpretations of S-percepts that give rise to mental imagery. S-percepts and A-percepts may be viewed as dual facets of a single entity, namely, information about a perceived object . The proposed architecture affords dual perspectives (symbolic and imaginal) on visual representations and similar mechanisms may underlie human visual perception and mental imagery.

REFERENCES

Anderson, J. R. (1978). Arguments concerning representations for mental images. *Psychological Review*, 85, 249-277.

Chandrasekaran, B., Goel, A., & Allemang, D. (1988). Connectionism and information processing abstractions: the message still counts more than the medium. *AI Magazine*, 9:4, 24-34.

Chandrasekaran, B., & Narayanan, N. H. (in press). The dual nature of visual representations. (Technical Report). Laboratory for Artificial Intelligence Research, Department of Computer & Information Science, Ohio State University, Columbus, OH.

Harnad, S. (1988). Mind, machine, and Searle. *Journal of Experimental and Theoretical Artificial Intelligence*, 1, 5-27.

Kosslyn, S. M., & Schwartz, S. P. (1977). A simulation of visual imagery. *Cognitive Science*, 1, 265-295.

Kosslyn, S. M., & Pomerantz, J. R. (1977). Images, propositions, and the form of internal representations. *Cognitive Psychology*, 9, 52-76.

Kosslyn, S. M. (1981). The medium and the message in mental imagery: a theory. *Psychological Review*, 88, 46-66.

Marr, D., & Nishihara, H. K. (1978). Representation and recognition of the spatial organization of three dimensional shapes. *Proceedings of the Royal Society*, 200, 269-294.

Pinker, S. (1980). Mental imagery and the third dimension. *Journal of Experimental Psychology: General*, 109, 354-37.

Pylyshyn, Z. W. (1981). The imagery debate: analogue media versus tacit knowledge. *Psychological Review*, 88, 16-45.

Pylyshyn, Z. W. (1984). *Computation and cognition: towards a foundation for cognitive science*. Cambridge, Mass.: MIT Press.

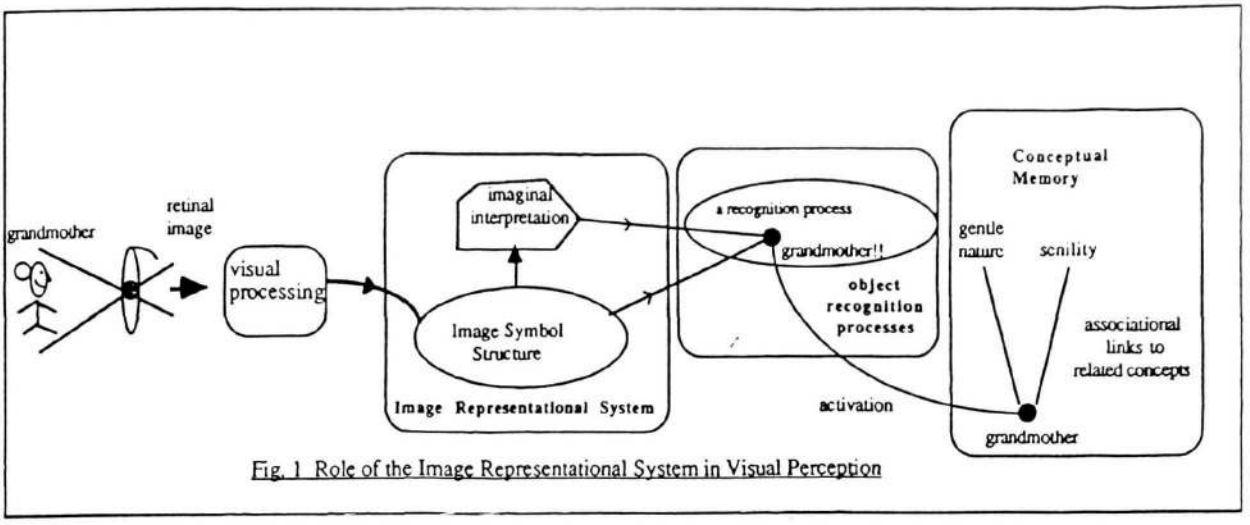


Fig. 1 Role of the Image Representational System in Visual Perception

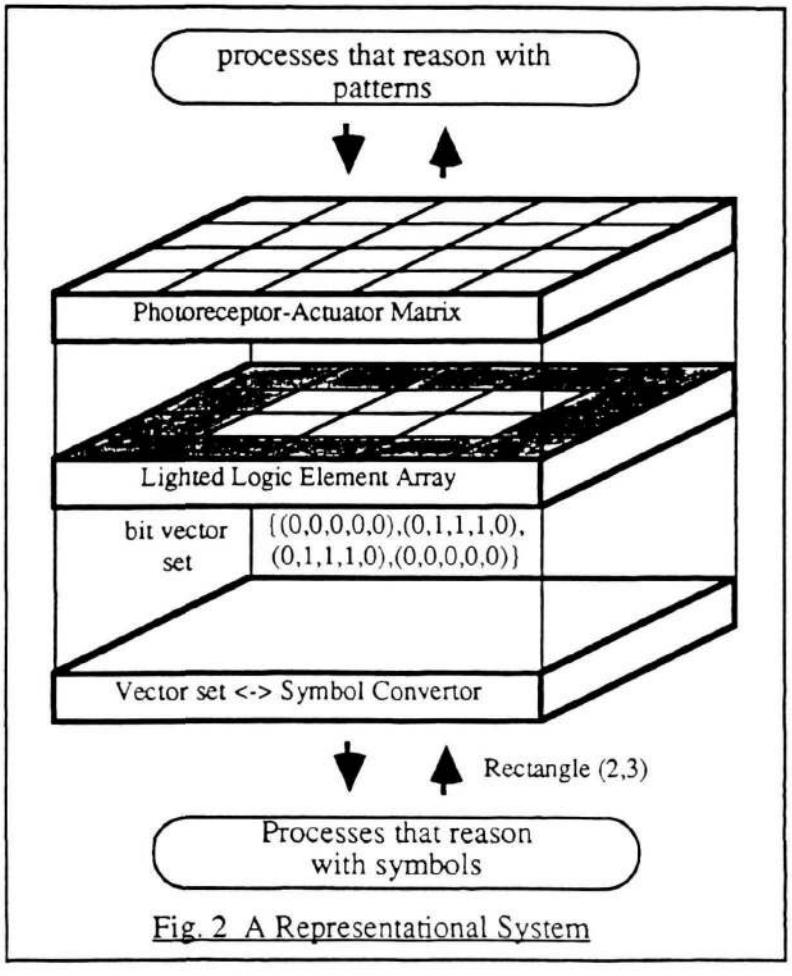


Fig. 2 A Representational System

Language Acquisition via Strange Automata.

Jordan B. Pollack

Laboratory for AI Research &
Computer & Information Science Department
The Ohio State University
2036 Neil Avenue
Columbus, OH 43210
(614) 292-4890
pollack@cis.ohio-state.edu

ABSTRACT

Sequential Cascaded Networks are recurrent higher order connectionist networks which are used, like finite state automata, to recognize languages. Such networks may be viewed as discrete dynamical systems (Dynamical Recognizers) whose states are points inside a multi-dimensional hypercube, whose transitions are defined not by a list of rules, but by a parameterized non-linear function, and whose acceptance decision is defined by a threshold applied to one dimension. Learning proceeds by the adaptation of weight parameters under error-driven feedback from performance on a teacher-supplied set of exemplars. The weights give rise to a landscape where input tokens cause transitions between attractive points or regions, and induction in this framework corresponds to the clustering, splitting and joining of these regions. Usually, the resulting landscape settles into a finite set of attractive regions, and is isomorphic to a classical finite-state automaton. Occasionally, however, the landscape contains a "Strange Attractor" (e.g fig 3g), to which there is no direct analogy in *finite* automata theory.

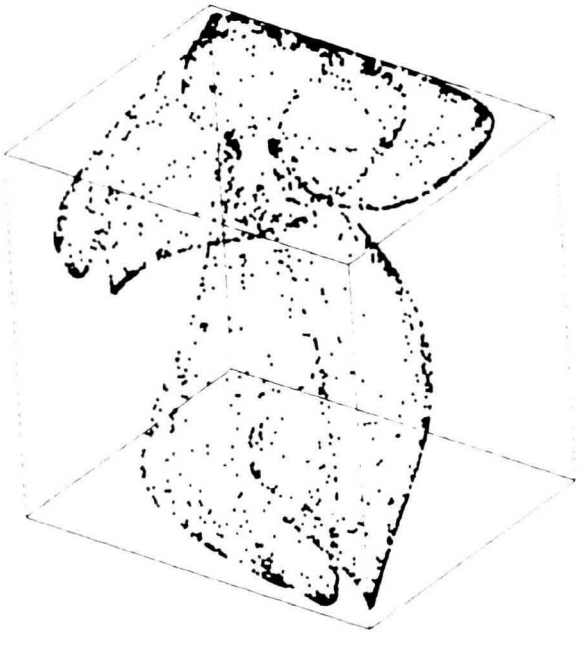


Figure 3g: *Infinite states of a "strange" automaton?*

1. Introduction & Background

Recently, J. Feldman (personal communication) posed the language acquisition problem, as a challenge to connectionist networks. In its most general form, it can be stated quite simply:

Given a language, specified by example, find a machine which can recognize (or generate) that language.

The problem is long-standing and has many specialized variants, especially driven by the goals of of various disciplines. On the one hand mathematical and computational theorists might be concerned with the basic questions and definitions of learning, or with optimal algorithms (Angluin, 1982; Feldman, 1972; Gold, 1967; Rivest & Schapire, 1987). On another hand, linguists may be concerned with how the question of learnability discriminates among grammatical frameworks and specifies necessarily innate properties of mind. On the third hand, psychologists might be concerned, in detail, with how a computational model actually matches up to the empirical data on child language acquisition. Rather than attempting to survey these areas, I point to the excellent theoretical review by (Angluin & Smith, 1983) and the books by (Wexler & Culicover, 1980) and by (MacWhinney, 1987) covering the linguistic and psychological approaches.

In this paper I expose a recurrent high-order back-propagation network to both positive and negative examples of boolean strings, and report that although the network does not find the minimal-description finite state automata for the languages (which is intractable), it does induction in a novel and interesting fashion, and searches through a hypothesis space which, theoretically, is not constrained to machines of finite state. This interpretation is dependent on an analogy among automata, neural networks, and non-linear dynamical systems. The pairwise sub-analogies are, of course, longstanding, as (McCullogh & Pitts, 1943) connected automata to neural nets, (Ashby, 1960) studied nets as dynamical systems, and (Wolfram, 1984) treated (cellular) automata as dynamical systems.

Although we usually think of the transitions among states in a finite-state automata as being fully specified by a table, a transition function can also be specified as a mathematical function of the current state and the input. For example, to get a machine to recognize boolean strings of odd parity, one merely has to specify that the next state is the exclusive-or of the current state and the input. Generalizing from a multilayer networks' ability to perform exclusive-or to the various constructive and existence proofs of the functional/interpolative power of such networks (Hornik et al., To Appear; Lapedes & Farber, 1988; Lippman, 1987), it is pretty obvious that recurrent neural networks can work just like finite state automata, where the transition table is folded up into some moderately complex boolean function of the previous state and current input.

From a different point of view, a recurrent network with a state evolving across k units can be considered a k-dimensional discrete-time dynamical system, with a precise initial condition, $z_k(0)$ and a state space in a bounded subspace of $\mathbf{R}^k$ (i.e., "in-a-box" (Anderson et al., 1977)). The input string, $y_j(t)$, is merely considered "noise" from the environment which may or may not affect the systems evolution, and the governing function, F, is parameterized by weights, W:

$$z_k(t+1) = F_W(z_k(t), y_j(t))$$

If we view one of the dimensions of this system, say z_a as an "acceptance" dimension, we can define the language accepted by such a *Dynamical Recognizer* as all strings of input tokens evolved from the precise initial state for which the accepting dimension of the state is above a certain threshold.

The first question to ask is how can such a dynamical system be constructed, or taught, to accept a particular language? The weights in the network, individually, do not correspond directly to graph transitions or to phrase structure rules. The second question to ask is what sort of generative power can be achieved by such systems?

2. The Model

To begin to answer the question of learning, I now present and elaborate upon my earlier work on Cascaded Networks (Pollack, 1987), which were used in a recurrent fashion to learn parity, depth-limited parenthesis balancing, and to map between word sequences and proposition representations (Pollack, To Appear). A Cascaded Network is a well-controlled higher-order connectionist architecture to which the back-propagation technique of weight adjustment (Rumelhart et al., 1986) can be applied. Basically, it consists of two subnetworks: The *function network* is a standard feed-forward network, with or without hidden layers. However, the weights are dynamically computed by the linear *context network*, whose outputs are mapped in a 1:1 fashion to the weights of the function net. Thus the input pattern to the context network is used to "multiplex" the the function computed, which can result in simpler learning tasks. For example, the famous Exclusive-or function of two inputs can be decomposed into two simpler functions of one input which are selected by the other:

$$XOR(y) = \begin{cases} y & \text{if } x = 0 \\ \neg y & \text{if } x = 1 \end{cases}$$

A simple 1-1 feedforward network (with 2 weights) can implement either of these functions. Identity is $g(3y - 1.5)$, and inversion is $g(1.5 - 3y)$, where $g(z) = 1/1+e^{-z}$, the usual sigmoidal squashing function. The essential idea of cascading is that these solutions in weight space can be a parameterized vector function of x.

Back-propagation is quite straightforward on a cascaded network. After determining the error terms for the weights of the function network, these are used as the error terms for the output units of the context network.

When the outputs of the function network are used as inputs to context network, a system can be built which learns to produce specific outputs for variable-length sequences of inputs. Because of the multiplicative connections, each input is, in effect, processed by a different function.

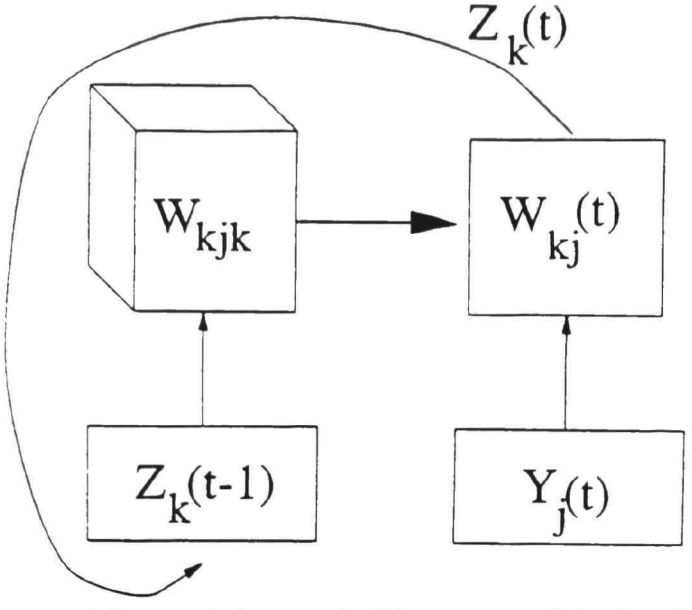


Figure 1. *A sequential cascaded network. The outputs of the function network are used as the next inputs to the context network, yielding a system whose function varies over time.*

Figure 1 shows a block diagram of a simple sequential cascaded network. Given an initial context, $z_k(0)$, and a sequence of inputs, $y_j(t)$, $t = 1...n$, the network computes a sequence of state vectors, $z_k(t)$, $t = 1...n$ by dynamically changing the set of weights, $w_{kj}(t)$:

$$w_{kj}(t) = w_{kjk}\, z_k(t-1)$$

$$z_k(t) = g(w_{kj}(t)\, y_j(t))$$

In previous work, I assumed that the teacher can supply a consistent and generalizable desired state for each member of a large set of strings. Unfortunately, this severely overconstrains the model. In learning a two-state machine like parity, this doesn't matter, as the 1-bit state fully determines the output. Such a teacher would be too powerful in the case of a higher-dimensional system, where we may know what the desired output of a system is but we *don't know* what its internal recurrent state should be.,

Jordan (1986) showed how recurrent back-propagation networks could be trained with "don't care" conditions. If there is no specific preference for the value of an output unit for a particular training example, simply consider the error term for that unit to be 0. This will work, *as long as that same unit receives feedback from other examples*. When the dont-cares line up, the weights to those unit will never change.

The first reaction, fully unrolling a recurrent network by maintaining vector histories (Rumelhart et al., 1986) has not lead to spectacular results (Mozer, 1988), the reason being that very tall networks with equivalence constraints between interdependent layers are unstable. My solution to this dilemma involves a *backspace*, unrolling the loop only once: After propagating the errors determined on only a subset of the weights by the known accept bit, d:

$$\frac{\partial E}{\partial z_a(n)} = (z_a(n) - d)\, z_a(n)\, (1 - z_a(n))$$

$$\frac{\partial E}{\partial w_{aj}(n)} = \frac{\partial E}{\partial z_a(n)}\, y_j(n)$$

$$\frac{\partial E}{\partial w_{ajk}} = \frac{\partial E}{\partial w_{aj}(n)} z_k(n-1)$$

The error on the remainder of the weights (e.g. $i = \{1 \cdots k \mid i \neq a\}$) is calculated by recycling the error on the accept plane with the network reset to its penultimate state:

$$\frac{\partial E}{\partial z_i(n-1)} = \sum_j \frac{\partial E}{\partial w_{ajk}} \frac{\partial}{\partial w_{aj}(n)}$$

$$\frac{\partial E}{\partial w_{ij}(n-1)} = \frac{\partial E}{\partial z_i(n-1)} y_j(n-1)$$

$$\frac{\partial E}{\partial w_{ijk}} = \frac{\partial E}{\partial w_{ij}(n-1)} z_k(n-2)$$

This is done, in batch (epoch) style, for a set of examples of varying lengths.

3. Experiments

Connectionist learning algorithms are very sensitive to the statistical properties of the set of exemplars which make up the learning environment. This has lead some psychological researchers to include the learning environment in the experimental parameters to manipulate (Plunkett & Marchman, 1989). Otherwise, it may not be clear if the results of a connectionist learning architecture are due to itself or due to skill or luck with setting up a collection of testcases. Therefore, I chose to work with test cases from the literature.

Tomita (1982) performed beautiful experiments in inducing finite automata from positive and negative examples. He used a genetically inspired hill-climbing procedure, which manipulated 9-state machines by randomly adding, deleting or moving transitions, or inverting the acceptability of a state, and accepting mutations based on their ability to improve the machine. Tomita ran his system on 7 cases and their complements. Each case was defined by two small sets of boolean strings, accepted by and rejected by the regular languages listed below.

1	1*
2	(1 0)*
3	no odd zero strings after odd 1 strings
4	no triples of zeros
5	pairwise, an even sum of 01's and 10's.
6	number of 1's - number of 0's = 3n
7	0*1*0*1*

For uniformity, I ran all 7 cases on a sequential cascaded network of a 1-input 4-output function network (with bias, 8 weights to set) and a 3-input 8-output context network with bias. The total of 32 weights is essentially arranged as a 4 by 2 by 4 array. Only three of the output dimensions were fed back to the context network, along with a set of biases, and the 4th output unit was used as the acceptance dimension. The standard back-propagation learning rate was set to 0.3 and the momentum to 0.7. All 32 weights were reset to random numbers between ±0.5 for each run. Termination was when all accepted strings returned output bits above 0.8 and rejected strings below 0.2. I changed initial conditions during the period of experimentation, and used an initial state of (.2 .2 .2) for cases 1, 3, and 4, and (.5 .5 .5) for the rest.

3.1. Results

Of Tomita's 7 cases, all but cases #2 and #6 converged without a problem in several hundred epochs. Case 2 would not converge, and kept treating negative case 110101010 as correct; I had to modify the training set (by added reject strings 110 and 11010) in order to overcome this problem. Case 6 took several restarts and thousands of cycles to converge, cause unknown. Presentation of the complete experimental data is in a longer report (Pollack, 1990).

However, none of the minimal-description regular languages were induced by the network. Even for the first language 1*, the network did not create an inescapable error state, so a 0 followed by a long string of 1's would be accepted by the network. If the network is not inducing the smallest consistent FSA, what is it doing?

4. Analysis

In my attempts at understanding the resultant networks, the first approach was to analyze their corresponding finite-state automata. The procedure was very simple. I ran the network as a generator, subjecting it to all possible boolean strings as input, and collecting first, the set of strings for which the acceptance dimension was past threshold, and second, the set of states

(points in 3-space) visited by the machine.

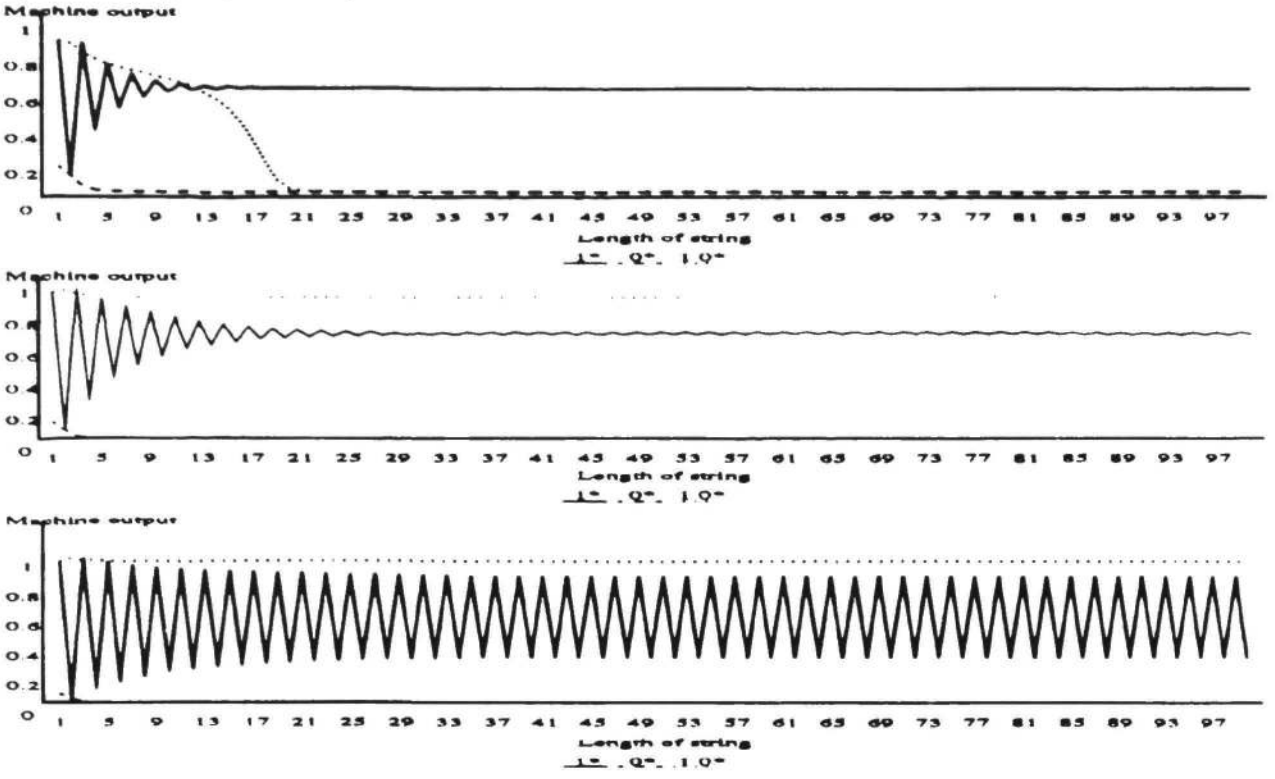


Figure 2. *Three stages in the adaptation of a network learning parity. (a) the test cases are separated, but there is a limit point for 1* at about 0.6. (b) after further training, the even and odd sequences are slightly separated. (c) after a little more training, the oscillating cycle is pronounced.*

Collecting the strings indicated one potential problem with the approach. After training the system can "fuzz out" for longer inputs than the ones given in the test cases. This can be examined for any particular recognizer. We simply observe the limit behavior on the accepting dimension for very long strings. For parity, since the string 1* requires an oscillation of states, we can examine the acceptance dimension as a function of the length. Figure 2 shows three stages in the adaptation of a network for parity. At first, despite success at separating a small training set, a single attractor exists in the limit, so that long strings are indistinguishable. After a little further training, the even and odd strings are separated, and after still further training, the separation is enough to set a threshold easily.

What initially appeared as a bug turns out to indicate a very interesting form of induction. Under feedback pressure to adapt, a slight change in weights leads to a point attractor being "bifurcated" into two. The result, in terms of performance, is significant! Before the split the network only worked correctly on short finite strings; afterwards, it worked on infinite strings.

4.1. Visualizing the Machines

Based upon preliminary studies of the parity example, my initial hypothesis was that a set of clusters would be found, organized in some geometric fashion to be harnessed by the way input causes the state to jump around. Thus, after collecting the state information, it seemed that this would cluster into dense regions which would correspond to states in a FSA. I wrote a diagnostic program to explore this space automatically, by taking an unexplored state and combining it with both 0 and 1 inputs. To remove floating-point fuzz, it had a parameter ε and threw out new states which were within ε euclidean distance from any state already known. Unfortunately, some of the machines seemed to grow exponentially in size as ε was lowered!

One reason for this seems to be that many "ravine" shaped clusters rather than point clusters are developed. Because the states are "in a box" of low dimension, we can view these machines graphically to gain some understanding of how the state space is being arranged. Graphs of the states visited by all possible inputs up to length 10, for the 7 test cases are shown in figure 3. Each figure contains 2048 points, but often they overlap.

The lack of closure under ε can now be seen as a completely different sort of attractor, making my earlier mapping attempt reminiscent of Mandelbrot's (1977, p. 25) essay about measuring the coastline of Britain. The variability in these structures certainly deserves further study, especially with regards to what types of landscapes are possible with different sized networks and alternative activation functions. The images (a) and (d) are what were expected, clumps of points which closely map to states of equivalent FSA's. Images (b) and (e) have simple ravines, which bleed into each other at their ends, probably indicating that longer strings will fuzz out.

Images (c), (f), and (g), are complex and quite unexpected, and will be further discussed below.

5. Related Work

The architecture and learning paradigm I used is closely related to recurrent archtecture devised by (Elman, 1988) and explored by others. Both networks rely on extending Michael Jordan's networks in a direction which separates visible output states from hidden recurrent states, without making the unstable "back-propagation through time" assumption. Besides our choice of language data to model, the two main differences are that

(1) They use a "predictive" paradigm, where error feedback is provided at every time step in the computation, and I used a "classification" paradigm, feeding back only at the end of the given examples.[1]

(2) They use a single layer (quasi-linear) recurrence between states, whereas I use a higher-order (quadratic) recurrence. It is certainly plausible that this quadratic nature allows more "radical" non-linearities to blossom.

Besides continued analysis, scaling the network up beyond binary symbol alphabets, immediate followup work involves comparing and contrasting our respective models with the other two possible models, a higher-order network trained on prediction, and a quasi-linear model trained on classification.

6. Discussion and Conclusion

The state spaces of the dynamical recognizers for Tomita cases 3, 6, and 7, are interesting, because, theoretically, they may be *infinite* state machines, where the states are not arbitrary or random, requiring an infinite table of transitions, but are constrained in a powerful way by some mathematical principle. I believe that it is closely related to related to Barnsley's work on iterated systems, where affine "shrinking" transformations direct an infinite stream of random points onto a underlying fractal or strange attractor.[2] In the recurrent network case, the "shrinking" is accomplished via the sigmoidal function, and the stream of random points are all possible input strings.

Certainly, the link between work in complex dynamical systems and neural networks is well-established both on the neurobiological level (Skarda & Freeman, 1987) and on the mathematical level (Derrida & Meir, 1988; Huberman & Hogg, 1987; Kurten, 1987). It is time that this link be further developed, especially as it applies to the question of the adequacy of connectionist, and other "emergent" approaches to high-level cognitive faculties, such as language (Pollack, 1989). The big question is whether any of the information structures which can be generated by complex dynamical systems can be at all correlated with the structures arising in natural language. Along these lines, (Crutchfield & Young, 1989) have analyzed the computation underlying period-doubling in chaotic dynamical systems and has found power equivalent to indexed context-free grammars.

In conclusion, I have by no means proven that a recurrent dynamical system can act as an efficient recognizer and generator for non-regular languages, though it does seem obvious.[3] But since Dynamical Recognizers are not organized as a PDA's or Turing Machines, it is not clear where the range of languages learnable by these systems would fit inside the Chomsky Hierarchy.

---

[1] Negative information is not crucial to the classification paradigm. Some distinction must be made among strings in order that the state space doesn't collapse into a point. The accept/reject bit is just the smallest such distinction.

[2] My use of the term "attractor" is more related to a stable pattern emerging from deterministic chaos (Lorenz, 1963) than to the traditional use (as energy minima) in optimization models (Ackley et al., 1985; Hopfield & Tank, 1985).

[3] Assuming rational numbers for states, a recurrent multiplicative relationship would be enough to start counting, which is necessary for beginning to handle context-free embeddings, of the sort $a^n b^n$; e.g. consider separate boolean inputs for a and b, and a recurrence $z(t+1) = .5a(t)z(t) + 2b(t)z(t)$. Assuming irrationals in the recurrence relationship, as physicists inadvertently do, and an ideal transcendental sigmoid, the "competance" languages may not even be computable.

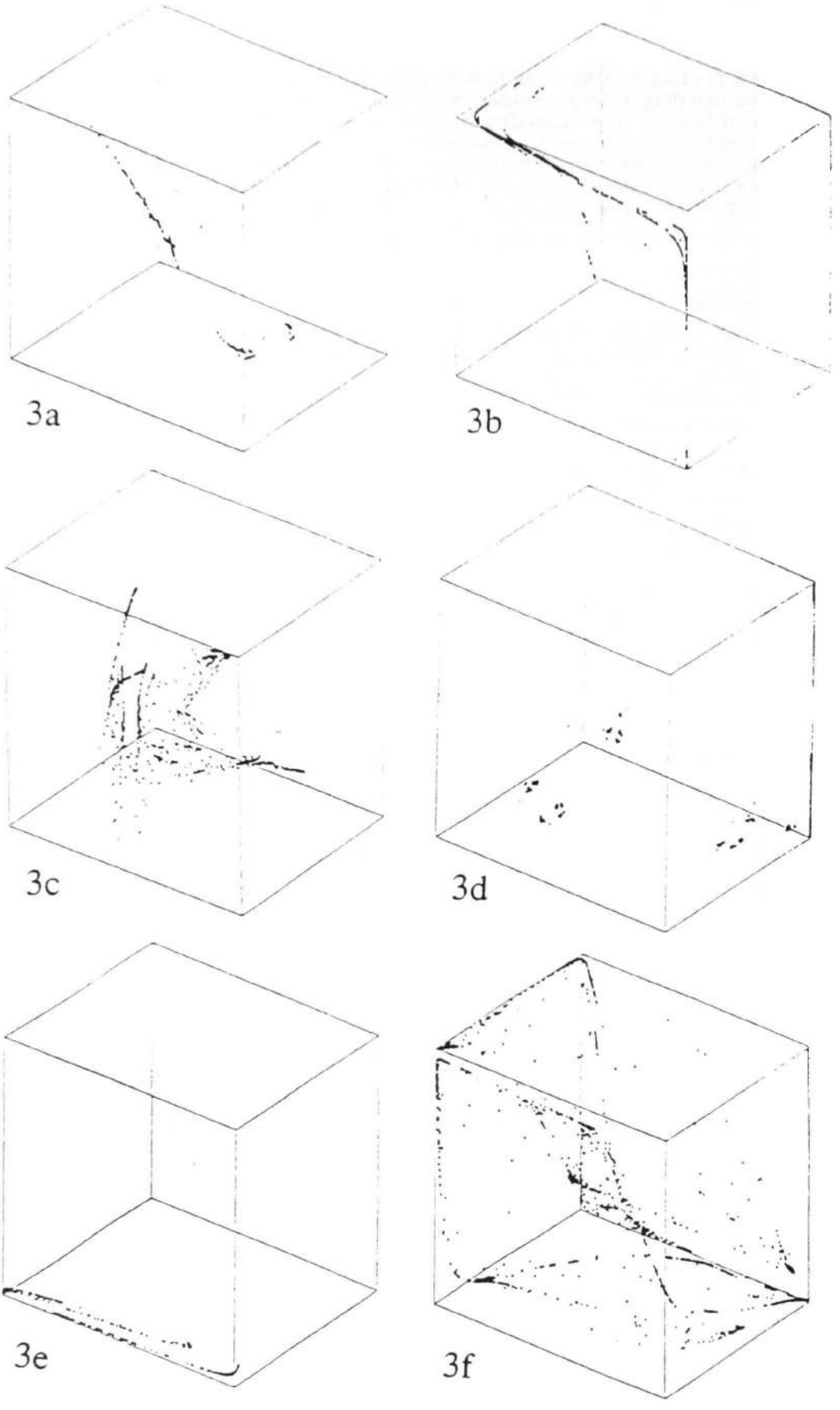


Figure 3. *Images of the attractors for six of the seven Tomita testcases. The points visited by all boolean input strings up to length ten are plotted. The seventh was viewed earlier.*

Nevertheless, we can consider the implications for language (and language acquisition) of a family of automata which smoothly evolve between finite and infinite state machines without massively duplicated transition tables: It will give rise to an induction method which will apply without a priori specification of the grammatical framework of a language in question. Generative capacity is neither natively assumed nor directly manipulated, but is an emergent property of the (fractal) geometry of a bounded non-linear system which arises in response to a specific learning task and is only revealed through performance.

This work is funded by Office of Naval Research Grant N00014-89-J-1200.

7. References

Ackley, D. H., Hinton, G. E. & Sejnowski, T. J. (1985). A learning algorithm for Boltzmann Machines. *Cognitive Science, 9*, 147-169.

Anderson, J. A., Silverstein, J. W., Ritz, S. A. & Jones, R. S. (1977). Distinctive Features, Categorical Perception, and Probability Learning: Some Applications of a Neural Model. *Psychological Review, 84*, 413-451.

Angluin, D. (1982). *Journal of the Association for Computing Machinery, 29*, 741-765.

Angluin, D. & Smith, C. H. (1983). Inductive Inference: Theory and Methods. *Computing Surveys, 15*, 237-269.

Ashby, W. R. (1960). *Design for a Brain: The origin of adaptive behaviour (Second Edition)*. New York: John Wiley & Sons.

Crutchfield, J. P & Young, K. (1989). Computation at the Onset of Chaos. In W. Zurek, (Ed.), *Complexity, Entropy and the Physics of INformation*. Reading, MA: Addison-Wesley.

Derrida, B. & Meir, R. (1988). Chaotic behavior of a layered neural network. *Phys. Rev. A, 38*.

Elman, J. L. (1988). Finding Structure in Time. Report 8801, San Diego: Center for Research in Language, UCSD.

Feldman, J. A. (1972). Some Decidability Results in grammatical Inference. *Information & Control, 20*, 244-462.

Gold, E. M. (1967). Language Identification in the Limit. *Information & Control, 10*, 447-474.

Hopfield, J. J. & Tank, D. W. (1985). 'Neural' computation of decisions in optimization problems. *Biological Cybernetics, 52*, 141-152.

Hornik, K., Stinchcombe, M. & White, H. (To Appear). Multi-layer Feedforward Networks are Universal Approximators. In *Neural Networks*. .

Huberman, B. A. & Hogg, T. (1987). Phase Transitions in Artificial Intelligence Systems. *Artificial Intelligence, 33*, 155-172.

Kurten, K. E. (1987). Phase transitions in quasirandom neural networks. In *Institute of Electrical and Electronics Engineers First International Conference on Neural Networks*. San Diego, II-197-20.

Lapedes, A. S. & Farber, R. M. (1988). How Neural Nets Work. LAUR-88-418: Los Alamos.

Lippman, R. P. (1987). An introduction to computing with neural networks. *Institute of Electrical and Electronics Engineers ASSP Magazine, April*, 4-22.

Lorenz, E. N. (1963). Deterministic Nonperiodic Flow. *Journal of Atmospheric Sciences, 20*, 130-141.

MacWhinney, B. (1987). In *Mechanisms of Language Acquisition*. Hillsdale: Lawrence Erlbaum Associates.

Mandelbrot, B. (1982). *The Fractal Geometry of Nature*. San Francisco: Freeman.

McCullogh, W. S. & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics, 5*, 115-133.

Mozer, M. (1988). A focused Back-propagation Algorithm for Temporal Pattern Recognition. CRG-Technical Report-88-3: University of Toronto.

Plunkett, K. & Marchman, V. (1989). Pattern Association in a Back-propagation Network: Implications for Child Language Acquisition. Technical Report 8902, San Diego: UCSD Center for Research in Languge.

Pollack, J. B. (1987). Cascaded Back Propagation on Dynamic Connectionist Networks. In *Proceedings of the Ninth Conference of the Cognitive Science Society*. Seattle, 391-404.

Pollack, J. B. (1989). Implications of Recursive Distributed Representations. In D. Touretzky, (Ed.), *Advances in Neural Information Processing Systems*. Los Gatos, CA: Morgan Kaufman.

Pollack, J. B. (1990). The Induction of Dynamical Recognizers. Tech Report 90-JP-Automata, Columbus, OH 43210: LAIR, Ohio State University.

Pollack, J. B. (To Appear). Recursive Distributed Representation. In *Artificial Intelligence*. .

Rivest, R. L. & Schapire, R. E. (1987). A new approach to unsupervised learning in deterministic environments. In *Proceedings of the Fourth International Workshop on Machine Learning*. Irvine, 364-475.

Rumelhart, D. E., Hinton, G. & Williams, R. (1986). Learning Internal Representations through Error Propagation. In D. E. Rumelhart, J. L. McClelland & the PDP research Group, (Eds.), *Parallel Distributed Processing: Experiments in the Microstructure of Cognition*, Vol. 1. Cambridge: MIT Press.

Skarda, C. A. & Freeman, W. J. (1987). How brains make chaos. *Brain & Behavioral Science, 10*.

Tomita, M. (1982). Dynamic construction of finite-state automata from examples using hill-climbing. In *Proceedings of the Fourth Annual Cognitive Science Conference*. Ann Arbor, MI, 105-108.

Wexler, K. & Culicover, P. W. (1980). *Formal Principles of Language Acquisition*. Cambridge: MIT Press.

Wolfram, S. (1984). Universality and Complexity in Cellular Automata. *Physica, 10D*, 1-35.

Miniature Language Acquisition: A touchstone for cognitive science

Jerome A. Feldman, George Lakoff, Andreas Stolcke and Susan Hollbach Weber
International Computer Science Institute, Berkeley CA

Abstract

Cognitive Science, whose genesis was interdisciplinary, shows signs of reverting to a disjoint collection of fields. This paper presents a compact, theory-free task that inherently requires an integrated solution. The basic problem is learning a subset of an arbitrary natural language from picture-sentence pairs. We describe a very specific instance of this task and show how it presents fundamental (but not impossible) challenges to several areas of cognitive science including vision, language, inference and learning.

1 Introduction

> touchstone (tuch' ston'). n. 1. a black siliceous stone used to test the purity of gold and silver by the color of the streak produced on it by rubbing it with either metal. 2. a test or criterion for the qualities of a thing.
>
> —Syn. 2. standard, measure, model, pattern.

Among the things that cognitive science has studied most are visual perception, language, inference, and learning [Posner, 1989]. However, these are often studied as if they were isolated from one another. Studies in visual perception rarely address the questions of how we perceive higher-order spatial relations and what systems of spatial concepts there are in the languages of the world. Computer vision and natural language processing are seen as different and unrelated disciplines. In psychology as well, language and vision are seen as distinct subspecialties, where specialists in one have little or nothing to do with the other.

The study of learning is, for the most part, just as isolated. Learning research often proceeds as if the content of what is learned did not matter. This is especially true of connectionist learning, which studies the learning of correlations among microfeatures, independent of content. One partial exception to this is language acquisition in the generative tradition, where a good deal of innateness is assumed [Pinker, 1989]. But there language acquisition is defined in a very limited way: "language acquisition" usually means just syntax acquisition and research has not attempted to characterize how people learn to describe what they see.

One way to start to unify several branches of cognitive science is to ask the question:

How could we learn to describe what we see?

We believe that addressing this question seriously could change the course of research in several subfields in a healthy way. We believe that these fields need to co-evolve, taking into account one another's constraints.

We realize of course that each of these fields is enormous and complex and largely unknown and that anything like a total integration at present is impossible. However, we believe that it is possible to undertake a small but nonetheless significant portion of that task now, and that doing so will have a sobering and an enriching effect on much of cognitive science.

We are proposing a new touchstone problem for cognitive science, a mini-task that is well-defined, very small relative to the overall job to be done, and yet significant enough so that one can learn a great deal. The Miniature Language Acquisition (MLA) task in its most general formulation is to construct a computer system such that:

The system is given examples of pictures paired with true statements about those pictures in an arbitrary natural language.

The system is to learn the relevant portion of the language well enough so that given a new sentence of that language, it can tell whether or not the sentence is true of the accompanying picture.

There are a number of attractive features in this general task. It is strictly behavioral and theory-free: nothing has been said about the theory or methodology that should be employed in the task. The problem is closed in the sense that one cannot appeal to some forthcoming result in a related domain that will complete the story — the system has to do the whole job. There is no stipulation of how much should be built into the system and how much learned. And the requirement that the same system should work for equivalent fragments of any natural language rules out *ad hoc* solutions. The issue now becomes one of feasibility: whether there is an instance of this task that is currently approachable but still rich enough to meet our programmatic goals.

Of course, the MLA task is not a model of human language acquisition or adult language learning. The semantic and pragmatic context of real human communication is far too complex to use as a basis for a miniature language acquisition task. Even the domain of idealized two-dimensional scenes, which is the simplest we could find, involves considerable complexity as some coming examples will illustrate. In fact, one of the greatest appeals of the MLA task is that deep questions in several areas of cognitive science appear in sharp form, even in the very limited L_0 domain.

2 L_0: a specific formulation

We have been investigating the particular task (which we call L_0) of language acquisition in the domain of simple two-dimensional scenes. In order to define the task precisely, we give a set of rules for L_0 (see Figure 1). These rules have no linguistic significance and, of course, are not available to the learning system. The sentences in the chosen natural language are drawn from a finite corpus capturing (as closely as possible) the meanings of the English descriptions specified by the L_0 grammar. The MLA task requires that the system work for equivalent fragments of any natural language. There may be a question of what would constitute an "equivalent fragment" of another language with a very different structure and a different system of spatial concepts. We (tentatively) characterize an "equivalent fragment" as one that can describe the same range of visual inputs.

The visual inputs to an L_0 system are scenes of (up to 4) idealized geometric figures. A candidate system is presented with a picture and with one or more sentences that are grammatical in the given language and are true of that picture. The order and possible repetition of training examples can be whatever the system's designer specifies. We explicitly allow an isolated noun phrase (NP) as a sentence fragment as this should simplify the initial stages of learning. After training on no more than half the examples (and hopefully many fewer) the system is tested by being presented with pairs consisting of a picture and a grammatical sentence which may be true or false about the companion picture. Obviously, the system succeeds to the extent that it produces the right

S = NP | NP VP
NP = DET NP1 | DET NP1 and DET NP1
VP = VI PP | VT NP
NP1 = OBJ | SHADE OBJ | SIZE OBJ | SIZE SHADE OBJ
PP = REL NP
VI = is | are
VT = touches | touch
DET = a
OBJ = circle | square | triangle
SHADE = light | dark
SIZE = small | medium | large
REL = REL1 | far REL1
REL1 = above | below | to the left of | to the right of

Figure 1: A specification of L_0.

---

answers. It is important to realize that the L_0 grammar is hidden from the system. It is known only to the generator of the training inputs, and the restrictions it embodies (though not these particular rules) must somehow be discovered by the learning system.

From the point of view of linguistics, the task has important attractions. First, there are already enough rich descriptions of spatial systems for various Indo-European and non-Indo-European languages for a start to be made [Rudzhka-Ostyn, 1988; Langacker, 1987; Casad, 1982; Casad and Langacker, 1985; Hershkovits, 1986; Janda, 1986; Talmy, 1983; Talmy, 1972; Talmy, 1985]. Second, the task, even in its simplest form, is demanding enough so that much deeper research on those spatial systems will be required. Third, the task focuses the attention of cognitive science on languages other than English, with special attention to the non-Indo-European languages, where the spatial systems are often very different from what English speakers are used to. Fourth, the task is semantically driven, which will require serious attention to the relation between syntax and semantics.

3 Variants

One advantage of the highly specific formulation of the L_0 task is that it focuses attention on what we consider to be the essence of the problem: acquiring syntactic descriptions of a limited but grounded semantics. The acquisition strategy should of course be extensible to a broader semantic range with the same grounding.

In addition to the base L_0 of Figure 1 we are looking at small variations. A specification that can be derived from Figure 1 by adding up to two words and two grammatical constraints is an acceptable variant of the task. One would not be happy with a system that worked for exactly L_0 but totally broke down for one of these minor variants. We think of a solution as *robust* if its designer can revise it to accommodate any single L_0 variation in one day. We are only interested in robust solutions, namely, those that are easily modifiable for an enormous range of minor variants in any natural language. This should guarantee that any robust solution must be doing quite a

few things right. As we shall see, adding even one minor variation can produce a great deal of complexity.

Note that the task as stated does not explicitly entail that linguistically significant generalizations be learned. We speculate that the robustness condition will guarantee a reasonable level of significant generalization. If a system can be extended simply to deal with any one of a very large number of variants, it would most likely have to have generalized pretty well.

Some of the L_0 variants that we have found useful to work with are:

1. Synonyms: e.g., have 'big' and 'large' used interchangeably.

2. Abstraction: add 'thing' to the definition of OBJ. This shows that one cannot simply identify an object with its shape.

3. Predicate negation: add "is not" and "are not" to the definition of VI. Negative sentences say much less about a picture, e.g. "A dark circle is not touching a square".

4. Sentence inversion: Add "PP VI NP" to the definition of S, e.g. "Above a circle is a small square". This forces the system to use grammatical cues in figuring out role assignment, rather than merely using relative position.

5. Plurals: add 'circles' and 'triangles' to the definition of OBJ, e.g. "A circle and a square are below large triangles".

6. Verb conjuncts: add "VT 'and' VI PP" to the definitions of VP, e.g. "A circle touches and is above a square". This makes explicit the requirement of allowing multiple references to a given object, e.g. "A circle touches a square and *the same circle* is above *the same square*".

7. Conjunctive attributes: add "REL1 'and' REL1" to the definition of REL. Conjunction can be subtle. For example, the sentence "A circle is above and to the left of a square" does not require that the circle be either 'above' or 'to the left of' the square.

8. Relative sizes: Add 'larger' and 'smaller' to definition of SIZE, e.g. "A circle is above a larger square". Expanding the scope of relational properties to comparatives highlights the necessity of adopting a visual representation that can handle abstract shape features and relations. One candidate is Ullman's [1984] visual routines. As Ullman points out, shape properties (e.g. connectedness) and relations (e.g. inside) can be computed by routines but are very hard to capture in a propositional semantics. Another L_0 example of this is the modifier 'far'.

9. Definite article: add 'the' to the definition of DET. Compare Figure 3 (c) with "The smaller circle is above the larger square".

10. Over and Under: add 'over' and 'under' to the definition of REL1. This is a far from trivial extension, since the term 'over' has dozens of related spatial senses [Lakoff, 1987]. For example, in Figure 2 (a), the circle is over (above) the square, while it is not in the topologically identical situation shown in Figure 2 (b). Another interesting asymmetry between 'under' and 'over' is depicted in Figure 3 (c).

11. Quantifiers: add 'every' and 'no' to the definition of DET, e.g. in Figure 2 (a), "No square is under a circle".

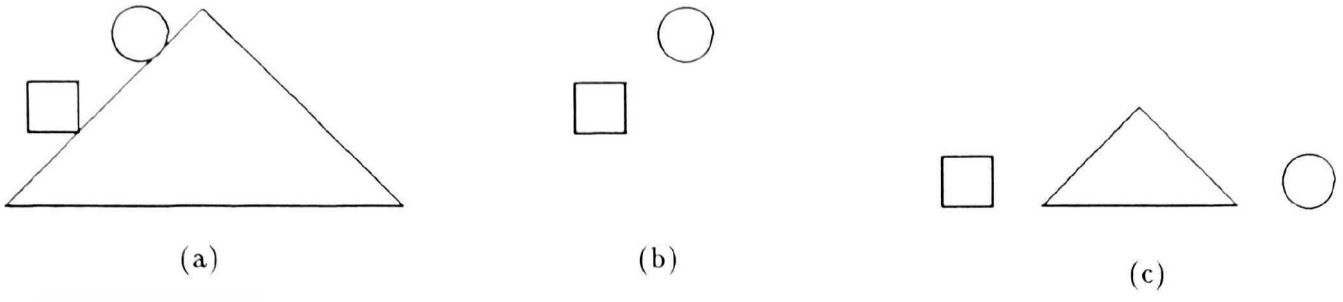


Figure 2: An interesting situation that highlights the complexity of the L_0 domain. (a) with the triangle as a frame of reference, the circle is seen to be 'over' the square; (b) without the apparent support of the triangle, the circle is no longer 'over' the square. Dialects differ on this. (c) The circle is over the triangle from the square.

Another possible set of variants involve motion and time and entail a whole range of new sentences and representations and inference issues. The six variants listed below should provide some of the flavor of the additional considerations.

1. Single object motion: add 'moved' to the definition of VT,

   and/or add "is now" to the definition of VI,

   and/or add 'was' to the definition of VI.

   Temporal variants raise the issue of object identity. It seems reasonable to assume that the system is given the inter-scene object correspondences.

2. Single object change: add "turned into" to the definition of VT.

3. Single object contact: add "bumped into" to the definition of VT.

4. Temporal non-change: add 'remains' to the definition of VI.

5. Pronoun: add 'it' to the definition of NP, eg. "It is now above it". Multiple scenes permit pronominal reference.

6. Trajectories: add "went between" to the definition of VT.

The examples involving motion and time suggest the need for an extended range of semantic primitives. For a variety of reasons [Feldman, 1988], we postulate that explicit trajectories will be an important primitive. It turns out that the trajectories are also useful in understanding some static scenes such as Figure 2 (c).

4 Conclusions

In order to succeed at the L_0 task, a program would have to be able to learn the grammars of at least small, simple fragments of any of the world's languages. It would have to be able to learn the

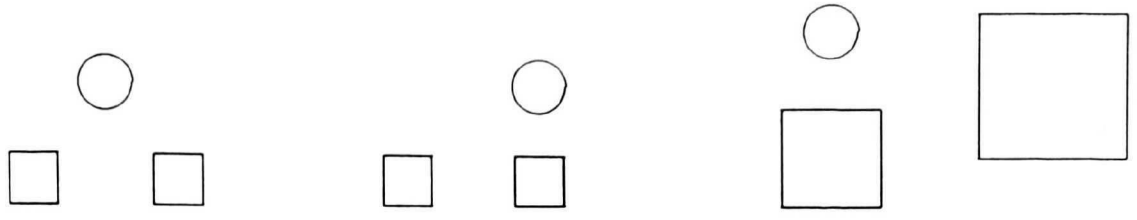


Figure 3: (a) the circle is 'over' the two squares, but the two squares are *not* 'under' the circle. (b) the circle is not over two squares (it is only over one of them). (c) The effect of context: "A smaller circle is above a larger square" is an acceptable description only with the indefinite article.

---

semantics of space for any of the world's languages. It would have to be able to learn to link that semantics of space to visual input. And it would have to be able to learn to do a range of spatial inferences for any of the spatial systems in the world's languages.

One question that would obviously arise immediately is just how much and what kind of innate structure would have to be built in. Building in too much would make it impossible to learn the task for some range of languages. Building in too little would most likely make the task intractable. Thus, if one has opinions about innate structure or the lack of it, this is a good task to test them on.

Miniature Language Acquisition, even at the L_0 level, is a cognitive science task that requires that very serious attention be paid to the enormous variety in syntax and semantics across the world's languages. The system will have to deal, for example, with classifier systems, which are common in the world's languages. It will have to deal with systems like Mixtec, where there are no prepositions, postpositions, or cases, but where spatial relations are characterized in terms of body-part projections and an elaborate spatial deixis system. And it will have to deal with languages like Cora, where there is an extensive deictic locative system which is based on image-superimposition [Langacker, 1987; Casad, 1982; Casad and Langacker, 1985]. This task will help cognitive science confront the fact that there is much more to human cognition than the concepts one finds in English or in familiar European languages.

Since the L_0 task is at the intersection of vision, language, inference, and learning studies, it encourages researchers in each of those fields to take the other fields seriously. It encourages linguists to take vision seriously and to ask how we can describe what we see. It encourages developmental psycholinguists to look beyond mere syntax acquisition and ask how we learn to describe what we see. It encourages vision researchers to focus on higher level systems of spatial relations of the sort that exist in the languages of the world. It encourages learning theorists to focus on content-specific learning.

Touchstone problems, which are easy to state but beyond current science, play an important role in many fields. Some examples are the "P=NP?" question in computability theory and the "copy demo" problem in robotics, and linguistic universals. We believe that the L_0 task can play such a role in contemporary cognitive science. Cognitive science is by nature interdisciplinary, but, unfortunately, it has developed subdisciplines that are going off in different directions without

paying enough attention to where they intersect and constrain each other. The adoption of the L_0 task as a touchstone would force these disciplines to pay attention to one another and, in the course of that, we think, change the disciplines in a healthy way.

References

[Badler, 1975] N. I. Badler, "Temporal Scene Analysis: Conceptual Description of Object Movements," Technical report, Technical Report TR-80, Department of Computer Science, University of Toronto, 1975.

[Casad, 1982] Eugene H. Casad, "Cora Locational and Structured Imagery," Technical report, Doctoral Dissertation, University of California at San Diego, 1982.

[Casad and Langacker, 1985] Eugene H. Casad and Ronald Langacker, "'Inside' and 'Outside' in Cora Grammar," *International Journal of American Linguistics*, 51:247–281, 1985.

[Feldman, 1988] Jerome A. Feldman, "Time, Space and Form in Vision," Technical report, TR-88-011, International Computer Science Institute, Berkeley CA., 1988.

[Harris, 1989] Catherine Harris, "A connectionist approach to the story of 'over.'," Technical report, Proceedings of the Berkeley Linguistics Society, 15, University of California, Berkeley CA, 1989.

[Hershkovits, 1986] Annette Hershkovits, *Language and Spatial Cognition; An Interdisciplinary Study of the Preposition in English*, Cambridge: Cambridge University Press, 1986.

[Herzog *et al.*, 1989] G. Herzog, C. K. Sung, E. Andre, W. Enkelmann, H.-H. Nagel, T. Rist, W. Wahlster, and G. Zimmermann, "Incremental Natural Language Description of Dynamic Imagery," In W. Brauer, editor, *Proceedings of the Third International GI Congress '89*. New York: Springer, 1989.

[Hildreth and Ullman, 1989] Ellen C. Hildreth and Shimon Ullman, "The Computational Study of Vision," In Michael I. Posner, editor, *Foundations of Cognitive Science*, pages 581–630. Bradford Books, MIT Press, Cambridge MA, 1989.

[Jackendoff, 1983] Ray Jackendoff, *Semantics and Cognition*, MIT Press, Cambridge MA, 1983.

[Janda, 1986] Laura Janda, "A Semantic Analysis of the Russian Verbal Prefixes ZA-, PERE-, DO-, and OT-," *Slavistiche Beitrage, Munich: Sagner*, 192, 1986.

[Lakoff, 1987] George Lakoff, *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*, University of Chicago Press, 1987.

[Langacker, 1987] Ronald Langacker, *Foundations of Cognitive Grammar Volume 1*, Stanford: Stanford University Press, 1987.

[Miller and Johnson-Laird, 1976] George A. Miller and Philip Johnson-Laird, *Language and Perception*, Harvard University Press, Cambridge MA, 1976.

[Pinker, 1989] Stephen Pinker, "Language Acquisition," In Michael I. Posner, editor, *Foundations of Cognitive Science*, pages 359–400. Bradford Books, MIT Press, Cambridge MA, 1989.

[Posner, 1989] Michael I. Posner, *Foundations of Cognitive Science*, Bradford Books, MIT Press, Cambridge MA, 1989.

[Rudzhka-Ostyn, 1988] Brygida Rudzhka-Ostyn, editor, *Topics in Cognitive Linguistics*, Philadelphia: John Benjamins, 1988.

[Rumelhart, 1989] David E. Rumelhart, "The Architecture of Mind: a Connectionist Approach," In Michael I. Posner, editor, *Foundations of Cognitive Science*, pages 133–160. Bradford Books, MIT Press, Cambridge MA, 1989.

[Siskind, 1990] J. M. Siskind, "Acquiring Word Meanings," forthcoming thesis, MIT, 1990.

[Smith, 1989] Edward E. Smith, "Concepts and Induction," In Michael I. Posner, editor, *Foundations of Cognitive Science*, pages 501–526. Bradford Books, MIT Press, Cambridge MA, 1989.

[Sopena, 1988] Josep Maria Sopena, "Verbal Description of Visual Blocks World Using Neural Networks," Technical report, Technical Report, Departament de Psicologia Basica, Universitat de Barcelona, 1988.

[Stolcke, 1990] Andreas Stolcke, "Learning Feature-based Semantics with Simple Recurrent Networks," Submitted to the 12th Annual Conference of the Cognitive Science Society, MIT, July 1990.

[Talmy, 1972] Leonard Talmy, "Semantic Structures in English and Atsugewi," Technical report, Doctoral Dissertation, University of California at Berkeley, 1972.

[Talmy, 1983] Leonard Talmy, "How Language Structures Space," In Herbert Pick and Linda Acredolo, editors, *Spatial Orientation: Theory, Research, and Application.* New York: Plenum Press, 1983.

[Talmy, 1985] Leonard Talmy, "Force Dynamics in Language and Thought," In *Papers from the Parasession on Causatives and Agentivity at the Twenty-First regional Meeting of the Chicago Linguistic Society*, pages 293–337, 1985.

[Ullman, 1984] S. Ullman, "Visual Routines," *Cognition*, 18:97–157, 1984.

[Vandeloise, 1984] Claude Vandeloise, "Description of Space in French," Technical report, Doctoral Dissertation. University of California at San Diego, 1984.

[Waltz and Boggess, 1979] David L. Waltz and Lois C. Boggess, "Visual Analog Representations for Natural Language Understanding," In *IJCAI-79*, pages 926–934, 1979.

[Weber and Stolcke, 1990] Susan Hollbach Weber and Andreas Stolcke, "L_0: A Testbed for Miniature Language Acquisition," Technical report, TR 90-010, International Computer Science Institute, Berkeley CA., 1990.

[Wexler and Culicover, 1980] Kenneth Wexler and Peter Culicover, *Formal Principles of Language Acquisition*, Cambridge, Mass.: M.I.T. Press, 1980.

[Winograd, 1971] Terry Winograd, "Procedures as a Representation for Data in a Computer Program for Understanding Natural Language," Technical report, MAC-TR-84, MIT, Cambridge, MA, January 1971.

Routine Evolution as the Microgenetic Basis of Skill Acquistion

Philip E. Agre
Department of Computer Science
University of Chicago

Jeff Shrager
System Sciences Laboratory
Xerox Palo Alto Research Center

Recent work in several areas has begun to demonstrate the value of microgenetic studies of the evolution of human activities. By paying close attention to the circumstantial detail that chronometrically based statistical methods suppress, microgenetic methods redirect our theoretical attention from putative mental mechanisms to the inherent logic of activities *in situ*. We have been studying in detail the actions of a single subject engaged in a routine photocopying job. Although her actions speed up according to a classical power law, the changes in her performance derive primarily from a constant and quite complex evolution of her patterns of interaction with her environment. Microgenetic analysis of the evolution of three aspects of the subject's activity demonstrates the dialectical relationship between the person-in-activity and the setting in which the activity takes place. Our goal is to build a theory of skill acquisition that does justice to the detailed organization and situational embedding of activity.

1 Introduction

If you do something over and over, you get better at it. A substantial literature understands this commonplace observation in terms of something called a *practice effect*. In this literature, "better" generally means "faster" and what gets faster is hypothesized to be mental processing. When this formulation is operationalized experimentally, one obtains a curve recording how long an experimental task has taken as a function of how much practice the subject has had with it. It is among the most robust results of experimental psychology that this curve, when averaged across subjects, approximates a power function, so that the subjects can be viewed as getting, say, thirty percent better on each iteration relative to some asymptote. This is called the "power law of practice."[1]

Many cognitive accounts of the power law and other practice effects have been proposed, including the shifting of procedural subunits from controlled to automatic processing (Shiffrin & Schneider, 1977), the substitution of procedural subunits by remembered results (Newell & Rosenbloom, 1981; Siegler & Shrager, 1984); and the strengthening and associated speedup or collapsing of procedural subunits (Anderson, 1983). The principal concern within this literature has been to isolate a speedup effect independent of complex changes in the activity. This has led most investigators to study practice effects by combining data from several subjects, thus suppressing the sharp changes in performance time that result from qualitative modifications of each individual subject's behavior.

A number of authors have pointed out that performance may improve as a result of changes in the way the subject carries out the task rather than by acceleration of mental processing. Crossman (1959) derived general laws of practice by statistical combination of different possible procedures for accomplishing a task. Anzai and Simon (1979) proposed a theory of skill acquisition incorporating a number of mechanisms including chunking, avoiding bad moves, and generalization. Cheng (1985) warned of the possibility of general "restructuring," a process she likens to changes in the brain's software. Siegler (1987) demonstrates the dangers of statistical combination of performance times in inadvertently hiding important individual differences in the procedures that compose skill, or even worse, of giving credit to a procedure that does not exist at all. Finally, Siegler and Jenkins (1989) observed the discovery of a new strategy in one child's arithmetic performance.

All of these projects begin from the assumption that speed-up is a unitary phenomenon. This assumption leads to a search for general mental speedup mechanisms, such as those based on strengthening, chunking, or restructuring. This strategy, we feel, limits these research programs in three ways. First, it leads some

[1] We recommend Newell and Rosenbloom (1981) for a careful history and analysis of the power law.

of them to suppress the qualitative structure of activities so as to focus on aggregate properties such as elapsed time. Second, even those authors who attend to the details of individual performance claim domain independence for their theories; as a consequence, the nature of "domains" is never clarified or given a substantive role in the theory. And third, the search for mental speedup mechanisms leads *all* of these researchers to treat activities and changes as wholly mental even though, as we will argue, they are not. All of the accounts hypothesize that what changes is some structure located primarily in the subject's mind, such as a plan, algorithm, or memory configuration.

To overcome these limitations, we propose a different research strategy for investigating the evolution of skills through practice. We propose taking the evolving organization of an activity to be a central phenomenon, not a marginal one. Specifically, we propose viewing individuals' activities as taking place through interactions with the world and not just in their heads. Activities derive their organization not only from mental processes and structures but from interactions with a potentially changing environment. The topic of study, then, is not "speed-up" or "performance improvement" as such but rather the qualitative processes through which routine activities evolve over time. We refer to this phenomenon as "routine evolution."

This strategy is consistent with a broad movement in computational and social theories toward viewing human activities as essentially situated in particular concrete circumstances (Agre, in preparation; Lave, 1988; Suchman, 1987). This is not to deny the existence of mental processing, but it is to give interactions with the environment a central role in organizing activity and in determining its course in particular cases. Lave (1988) has recommended adopting as the units of analysis the *person-in-activity* and the *setting* in which the activity takes place. These two entities participate in a dialectical relationship with one another. The term "dialectical" has three implications: that the person-in-activity and setting interact with one another; that they reciprocally influence one another; and that they cannot sensibly be defined except with reference to one another. On this view, routine evolution is best viewed as reflecting the evolution of the person/setting dyad and not simply of mental contents or physical environments on their own. Thus the challenge for research is how to study this unfolding dialectic of interaction as a natural, qualitative, self-organizing process.

2 Routine Evolution in Photocopying

In pursuit of a theory of routine evolution, we are conducting detailed analyses of the evolution of particular routine activities performed by particular individuals. In this paper we describe one of our early studies. We chose to observe routine evolution in the activity of photocopying because it is a very common office task, making it easy to find non-expert subjects with some experience. Indeed, it is common enough that one might not even expect to find practice effects at all.

We employed a deceptive experimental paradigm so that the subjects' awareness of being observed would not lead them to try self-consciously to optimize their performance. We advertised at Stanford University for a temporary office worker to help out in preparations for an upcoming conference. No mention was made of a psychology experiment. When the subject, whom we shall call Sarah, arrived she was shown to a room containing a Xerox 1075 copier (Figure 1). We had mounted a camera in the ceiling of the room, overlooking the entire copier, and had hidden a microphone near the copier. The camera was not easily visible, and in the debriefing Sarah reported having no awareness that she was being observed. Upon completion of the task, Sarah received $20, after which the experimental cover was lifted and she was debriefed. She was also given the option of having the videotape erased immediately and on the spot without any questions asked, and without pressure from the experimenter, to the extent that this was possible.

There were four sorts of copying task, but only one of these concerns us here. In this task, three papers were selected from a collection of readings from a Stanford psychology course, bound by a 0.5 inch black "Cerlox" binder, with front and back covers of heavy green paper. A post-it on the book's cover read: "Need 3 copies of the indicated papers—any style but stapled would be nice." The first of the three papers was 17 pages in length. The copy of this paper in the spiral-bound readings collection begins on page 67 (a right

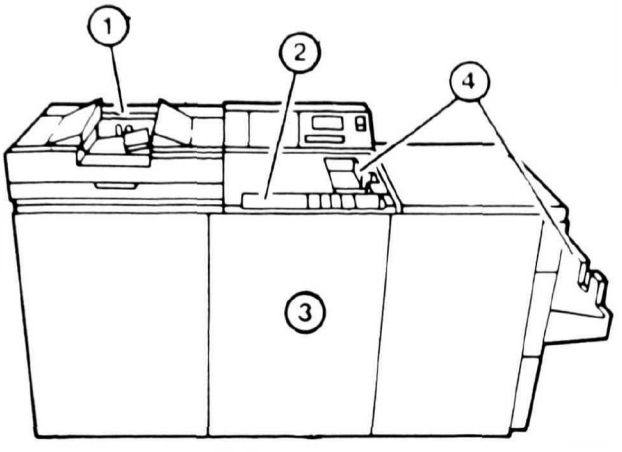


Figure 1: The Xerox 1075 photocopier: 1. Recirculating Document Handler (RDH), 2. Control Panel, 3. Copier Body, 4. Output Trays. The START button is the leftmost on the control panel, and the copier glass is on the copier body centered under the RDH.

hand page) of the 296 page book. Sarah began to copy this paper after about one hour and six minutes of work, and finished about four minutes later.

We begin our analysis by looking at the chronometric aspects of Sarah's performance. As we will see in a moment, Sarah's activity in this task is quite complex and changes constantly. Is she actually improving in terms of speed? Figure 2 shows the time in hundredths of seconds between odd-numbered presses of the copier's START button. We depict the time for successive pairs of pages because left-page-right-page is the natural cyclical unit in copying a bound document. We can see that although she begins fairly slowly, she rapidly increases in speed. In fact, this plot is very close to a power law. We hypothesize that if we average such data over a number of subjects, we will be able to produce an excellent classical power law of practice.

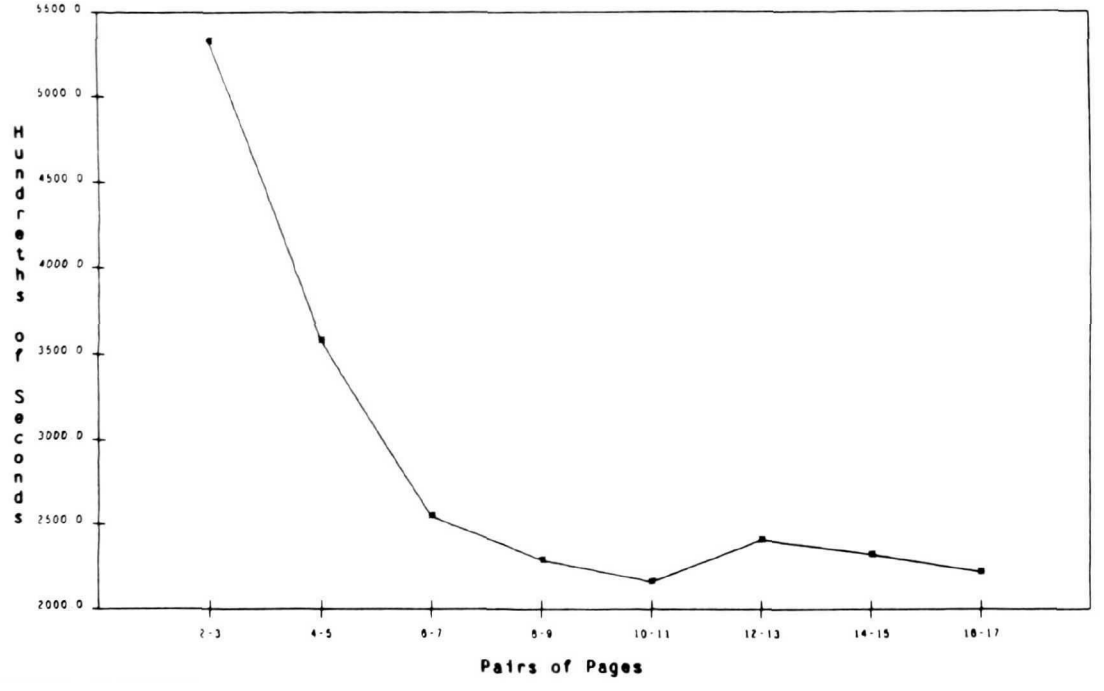


Figure 2: The elapsed time for each successive pair of copies.

3 Microgenetic Analysis

Having demonstrated that there is indeed speedup in this simple activity, and that it appears to be compatible with the traditional practice effect findings, we now want to locate the source of this effect. Chronometric methods cannot register the fine qualitative structure in the evolution of Sarah's routine for the copying task. Instead, we follow Vygotsky's advice (1978: 58–75; Scribner, 1985) to study the subject's activity as a historical phenomenon, finding its essence in its processes of change rather than in the "fossilized" outward structure that it might exhibit if allowed the time to become wholly mechanical. Specifically, we employ microgenetic methods such as employed by Siegler and Jenkins (1989). These methods involve the detailed documentation and coding of particular stretches of activity, a sort of micro-longitudinal tracking of the routine evolution process across a long period. The analysis we discuss in this paper covers the four minutes of Sarah's copying activity, observed at video-frame granularity of 1/30th of a second.

We will develop our argument by recounting a series of moments in the evolution of Sarah's photocopying routine. In doing this, we have two goals. First, we wish to demonstrate the utility of microgenetic analysis in making sense of the intricacies of the evolution of this activity. And second, we wish to demonstrate that even when activity is routine, it is continually self-organizing through interaction between the person-in-activity and the setting. A prerequisite of routine activity is the regularity of the relevant aspects of the environment, and this regularity is itself a product of the routine activity.

Sarah's actions while copying the first few pages are markedly circumspect. She settles the book carefully into place, readjusts the controls, looks around her a great deal, and checks the results. None of these aspects of her activity is apparent once her routine has settled down in the later part of the task. It is worth distinguishing this generic circumspection from actions which are present in the beginning but which later prove inessential. Some actions drop out; for instance, she does not close the RDH[2] cover after the third page. This change saves her about 6.3 seconds per cycle.[3] In addition, aspects of the activity slide on top of one another in time and begin to merge together. For example, her pressing of the START button—initially a clearly separate step—merges with other actions such as walking to the right of the copier or continuing to position the book with her left hand.

Sarah uses aspects of the setting to coordinate her activity, but the relationship between these changes over time in the task. For example, because Sarah is making three copies in this task, the copier flashes very brightly three times (at 0.8 second intervals) after the START button has been pressed and before the book can be set up for the next page. Over the course of the task these flashes grow into a pivotal resource in the organization of her activity. This development has two related aspects. First, a period of dead time of about 2.4 seconds intervenes between pressing START and the point at which the book can be moved. Second, the precise moment when manipulation of the book can resume can be identified by means of the third flash. Near the beginning of the task a moment passes between the third flash and Sarah's turning toward and setting up the next page. As time goes on, however, she begins setting up *even before* the third flash (that is, after the first or second flash) so that she is ready to go as soon as the third flash arrives. In the course of the task, this change reduces the time per cycle by about 5.7 seconds.

Having surveyed these general points let us focus on the evolution of a particular segment of each cycle, namely the period after each odd-numbered page, when Sarah turns to the next page. At this point in the cycle the right hand page is face down on the copier glass and the left hand page is draped over the right edge of the glass. She needs to turn the book face up, put it down on the copier, turn the page, and replace the book on the glass so that the next (left hand, even-numbered) page is face down on the glass. In each case, Sarah proceeds by a rapid and fluent sequence of hand-over-hand moves that takes some patience to record accurately. The specific way in which she accomplishes these moves provides a clear example of the

[2] "Recirculating Document Handler" – see Figure 1.

[3] We provide these approximate measurements purely as part of the analysis of the source the power law shown in Figure 2. It is always *extremely* difficult to measure in a principled way the chronometric contribution of these aspects of the activity. In this case, for example, closing the RDH leaves Sarah's right hand closer to the START button than when she is touching the book and does not close the RDH. Therefore, although some time is lost in RDH manipulation, her hand spends less time traveling to the START button. Any chronometric operationalization would have to find a systematic way of correcting for such effects.

dialectical character of routine evolution.

Table 1 catalogs which hand Sarah uses to accomplish each of eight functions in the page turning activity on each of the eight cycles of the copying task (R is right, L is left). It is important to understand that these "functions" are gross categories that we, the theorists, are picking out from Sarah's behavior. We do not know if she would describe the activity in these terms, and of course much more is going on in the page turning activity than this table captures.

With that in mind, we can describe the evolution of Sarah's page turning routine in three parts, as depicted in Table 1. Part A consists of the three functions: lifting the book to turn it face up, flipping it face up, and something we will call "assistance" that generally consists of grasping the free-hanging side of the book and pulling it around in the appropriate direction. Part B consists of lifting the page to be turned and flipping the page to the opposite side. Part C consists of lifting the right side of the now face up book, and flipping it face down. This operation too requires assistance with the free-hanging side of the book. Note that at the end of parts A and C, both of Sarah's hands are grasping the book and she uses both hands to settle the book in place, either face up as in part A (to turn the page) or face down as in part C (to continue copying), on the copier glass.

Last Page Copied

Functions	1	3	5	7	9	11	13	15	
lift to flip face up	R	R	R	R	R	L	R	L	part A
flip book face up	L	L	L	L	L	L	L	L	part A
flip assist	R	R	R	R	R	R	R	R	part A
lift page	L	R	L	R	R	R	R	R	part B
turn page	L	R	L	L	L	L	L	L	part B
lift to flip face down	R	R	R	R	R	R	R	R	part C
flip book face down	L	L	L	L	L	L	L	L	part C
flip assist	R	R	R	R	R	R	R	R	part C

Table 1: Which hand was used in each of eight functions in the page turning activity over the session.

At this level of analysis, Sarah uses the same hands in part C throughout the task. Her manipulation takes place in the sequence: right hand (lift), left hand (flip), right hand (assist). We will abbreviate this as "RLR". The fact that we do not see change in this sequence at this level of analysis should not be interpreted as absolute stability of this part of the activity. Although Sarah uses the same hands for these functions in the same sequence, there is considerable adjustment involved, for instance, in repositioning her hands and the book in order to accomplish smooth grasps.

Contrast the stability of part C with the relative instability of part A, which goes from a fairly stable RLR sequence to LLR in the page turns following pages 11 and 15. Finally, notice that part B begins inconsistently in the first three cycles (the left part of Table 1), moving from LL to RR to LL, but that in the last five cycles (the right part of Table 1) it has settled down into an RL pattern.

Our task, then, is to explain three contrasting phenomena: the unsystematic change in part A, the systematic change toward stability in part B, and the lack of change in part C. Let us consider these one at a time and then attempt to synthesize what we have learned.

Part A, turning the book face up, begins after the copier has finished copying a given odd-numbered page. As we have mentioned, Sarah has an evolving relationship with the third flash of light, which marks this point. Over a few cycles she learns to anticipate this flash, so that her hands are on the book ready to go when the flash arrives. Later, though, the dead time during the copier's flashing is sometimes filled by other activities. During the flashes for pages 5, 7, and 9, she walks to the right (off the bottom of the video

image), perhaps to check the copies in the side output tray. During pages 11 and 15, perhaps satisfied with the state of the output tray, she stands with her left hand resting on the spine of the book during the flashes. And during the flashes for page 13 she appears to take a moment to relax: she steps back from the copier, looks about, touches the copies sitting on the top of the copier, pulls her sleeve up slightly, and walks back toward the copier, arriving at the precise moment of the third flash. The variation in this segment of the activity is at least partially systematic, in her increasingly accurate anticipation of the third flash, but it is also influenced by a variety of other matters, such as her concerns (whatever they might have been) about the copies in the output tray. As a result of these developments, her physical relationship to the copier at the beginning of part A changes from cycle to cycle. The change is small at first but then grows larger as more elaborate activity comes to fill the dead time. She picks up the book at the beginning of step A with (approximately speaking) whatever hand is closer to its right edge. Which hand is closer, in turn, depends on her lateral (i.e., left or right) position in front of the copier, which in turn depends on what has been going on before to affect that position. Thus, in Table 1, we see that she starts out using RLR hand sequences to turn over the book but that this pattern breaks down later on. The alternation between RLR and LLR does not appear to be systematic, though she might have settled into an approximately stable LLR pattern if the document had more pages.

The evolution of part B is conditioned by several other factors. As part B begins, she has just finished turning the book face up (part A). In particular, she has just finished laying the book down, so that it is now in a highly standardized relationship to her body and especially to her hands. Her way of turning over the page evolves in a systematic way. As she is finishing with part A, her right hand is located at the top right corner of the book and her left hand is located along the bottom edge of the left half of the book. Her task at this point, of course, is to flip over the right-hand page. On the first such page, she begins lifting the top right corner of the page with her right hand but then also begins lifting the bottom of the page with her left hand. She then finishes turning the page with her left hand. On page 3, she begins by moving her right hand to the bottom right hand corner of the book and her left hand to the bottom of the left page of the book, as if smoothing the left page. She turns the page using her right hand, with token guidance from her left hand. On page 5, she moves her left hand to the bottom of the right page and turns the page entirely with her left hand. Meanwhile, her right hand remains on the upper right hand corner of the right page and appears to make a vestigial attempt to turn up the corner just before her left hand turns the page. On page 7 she settles into the emerging pattern of moving both hands to the bottom of the right hand page, lifting the page at the lower right hand corner with her right hand and then turning it over with her left hand. As her left hand turns the page, her right hand pauses for 0.4 seconds and then proceeds to the right edge of the book to prepare to grasp it for part C. Page 9 is very similar to page 7, which the exception that her right hand does not pause between handing off the turned-up page and proceeding to the right edge of the book. All of the remaining page turns look identical to this one, though in some of them it takes her an extra moment to get the page turned up, presumably due to pages sticking together. What we see here is a highly logical evolution of her page-turning routine. Each step changes incrementally from the previous one until the routine settles into a pattern that is stable so far as the representation in Table 1 is concerned. It is worth noting that none of the previous tasks in the experiment had called for her to turn pages in a bound book. She has presumably turned many pages in books before, but the particular way she has evolved to turns pages here appears well-adapted to several aspects of the physical setting: her standing posture, the height of the machine, the lateral placement of the book relative to her body, the mechanical properties of the Cerlox binding, and the affordance offered by the right edge of the book to the grasping operation which will commence part C.

In part C, Sarah uses the same hands for the same functions on every page. Parts A and C involve highly analogous tasks of flipping over the book, yet part A changes and part C does not. The reason for this discrepancy is plain in looking at the tape and from the analysis above. Whereas the relationship of Sarah's body to the setting changes before part A due to different activity during the flashes, the process of flipping the book over (part B) repositions the book with respect to her body. The processes of flipping the book over and turning the page have left her, again, in a highly standardized physical relationship to the book. Table 1 does not register the single exception, which occurs on page 13, when the flipping-over operation

in part A has left the book lying entirely on the copier glass, so that its right edge does not readily afford grasping. In this case, she quickly pushes the book to the right with her left hand before grabbing it with her right and proceeding with the regular RLR pattern.

Before drawing conclusions about the evolution of routine activity, let us gather some observations. Part A of the activity differed from parts B and C in that its initial conditions were subject to variation as a side-effect of previous activity (waiting on the copier). Parts B and C enjoyed very stable initial conditions due to the regularities in outcome of the previous operations (namely those of parts A and B, respectively). Part B differed from part C in that it underwent a steady, incremental process of evolution. Part C, by contrast, did not evolve. But as part B evolved, the "joint" between parts B and C evolved as well, as Sarah's hand began heading toward the right edge of the book immediately after handing the page off to her left hand. In general, the various parts of the evolving activity stay knitted together despite their various wanderings.

We have spoken of activity in terms of a dialectical relationship between a person-in-activity and the setting. Let us now substantiate this characterization in the present case in terms of the three aspects of a dialectical relationship that we listed before. Sarah and the copier plainly interact with one another: Sarah's actions are contingent in fine detail on the physical situation, upon which Sarah acts in turn. Moreover, Sarah and the copier exert a mutual influence on one another, as evidenced in the copier's evolving physical state and Sarah's evolving skill. And as this mutual influence proceeds, it becomes impossible to characterize either the copier's state or Sarah's skill except in terms of their reciprocal relationship. As part of this process Sarah's actions exhibit a growing fluency and a continually increasing adaptation to the specific details of the setting.

The conditions for stability of Sarah's activity are naturally analyzed in terms of this interaction. Parts B and C of this activity, unlike part A, could stabilize because their initial conditions were stable. One possible explanation of why part B evolved whereas part C did not is that part C, unlike part B, corresponds to a task that she has had to perform throughout the copying session, so that it is likely to have become more highly evolved. On the other hand, the difference may simply reflect the analytic filtering imposed by the representation in Table 1. In any event, the stability we observe is the stability of the person/setting dyad; changes are changes in this dyad. Even changes to Sarah's skill, which might be construed as mental changes, cannot readily be defined except in terms of the shifting patterns of interaction in which they participate. The subject's actions exhibit an increasing fluency and coherence, but these are emergent properties of a more complex process whose logic is the logic of photocopying.

4 Concluding Remarks

Practice is for us not a single "effect," or ten effects, but the concrete unfolding of a given activity in accord with its own logic. Such a view carries with it a severe methodological challenge: to find a way of studying human activity that both treats episodes of activity as singular events and facilitates the construction of accounts that have broad application. The microgenetic method is substantially more useful than the chronometric method for these purposes because it makes the qualitative organization of the learning process much more evident.

The microgenetic method thus enables us to overcome the three limitations of mentalistic accounts that we discussed at the outset. First, the detailed twists and turns of practice curves which are often treated as deviations from a general trend are, for us, the principal phenomena. The general trend that shows up as the smooth curve one obtains by averaging many individual performance is just that, a general trend, and not a reflection of a single underlying factor. Second, our analysis requires unpacking the detailed organization of the activity and the logic of its change. It thus calls for an analysis of the positive structure of the domain. Indeed, the notion of a domain of activity requires further analysis in terms of its broader social structuration, as suggested by Lave (1988) and the activity theorists (Davydov & Radzikhovskii, 1985). And third, in contrast to theories that locate skill and its acquisition in mental processes, we would locate activity and its evolution in the dialectical relationship between persons-in-activity and settings. Our data provide

clear motivation for such a view. We have not simply added a setting to the mind; we have enriched the analysis of each by taking account of its reciprocal involvement with the other.

This alternative understanding of skill acquisition makes less natural the reification of particular aggregate properties of the activity such as "speed" and "performance" or objective global principles such as a law of minimum effort. But then an important new question arises. These reified concepts had functioned to provide an account of the macroscopic coherence of the development of skills. Given that the subject's activity could have evolved along many possible paths, why does it evolve in the way it does? In other words, what kind of organizing principle *does* give skill acquisition its overall coherence? Routine evolution is the unfolding of the logic of a particular activity, but this logic itself is mediated by cultural constructions of the activity and of the considerations that might bear on it (piece-work versus hourly wages, precision, comfort, safety, aesthetics, relationships with other people, and so forth). A theory of practice effects will ultimately entail an understanding of the cultural meanings of practice and of skill.

References

Agre, P. E. (in preparation). *The dynamic structure of everyday life.* Cambridge University Press.

Anderson, J. R. (1983). *The architecture of cognition.* Harvard University Press.

Anzai, Y., & Simon, H. A. (1979). The theory of learning by doing. *Psychological Review, 86(2)*, 124-140.

Cheng, P. W. (1985). Restructuring versus automaticity: Alternative accounts of skill acquisition. *Psychological Review, 92(3)*, 414-423.

Crossman, E. R. F. W. (1959). A theory of the acquisition of skill-speed. *Ergonomics Journal*, 153-166.

Davydov, V. V., & Radzikhovskii, L. A. (1985). Vygotsky's theory and the activity-oriented approach in psychology. In J. V. Wertsch (Ed.), *Culture, communication, and cognition: Vygotskian perspectives.* Cambridge University Press.

Laird, J., Rosenbloom, P., & Newell, A. (1986). *Universal subgoaling and chunking: The automatic generation and learning of goal hierarchies.* Boston: Kluwer Academic Publishers.

Lave, J. (1988). *Cognition in practice: Mind, mathematics, and culture in everyday life.* Cambridge University Press.

Newell, A., & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Scribner, S. (1985). Vygotsky's uses of history. In J. V. Wertsch (Ed.), *Culture, communication, and cognition: Vygotskian perspectives.* Cambridge University Press.

Shiffrin, R. M., & Schneider, W. (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending, and a general theory. *Psychological Review, 84(2)*, 127-190.

Siegler, R. S. (1987). The perils of averaging data over strategies: An example from children's addition. *Journal of Experimental Psychology: General, 116(3)*, 250-264.

Siegler, R. S., & Jenkins, E. (1989). *How children discover new strategies.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Siegler, R. S., & Shrager, J. (1984). Strategy choices in addition and subtraction: How do children know what to do? In C. Sophian (Ed.), *Origins of cognitive skills.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Suchman, L., (1987). *Plans and situated actions.* Cambridge University Press.

Vygotsky, L., (1978). *Mind in society: The development of higher psychological processes* (M. Cole, V. John-Steiner, S. Scribner, and E. Souberman, Eds.). Cambridge, MA: Harvard University Press.

Philip E. Agre, Department of Computer Science, University of Chicago, 1100 East 58th St., Chicago, IL, 60637, (312)702-6209, agre@gargoyle.uchicago.edu; Jeff Shrager, Xerox Palo Alto Research Center, 3333 Coyote Hill Rd., Palo Alto, CA, 94304, (415)494-4338, shrager@xerox.com

The Disruptive Potential of Immediate Feedback

Lael J. Schooler

John R. Anderson

Carnegie-Mellon University

Abstract

Three experiments investigate the influence of feedback timing on skill acquisition in the context of learning LISP. In experiment 1 subjects receiving immediate feedback went through the training material in 40% less time than did those receiving delayed feedback, but learning was not impaired. A second experiment involved the use of an improved editor and less supportive testing conditions. Though subjects in the immediate condition went through the training problems 18% faster than did those in the delay condition, they were slower on the test problems and made twice as many errors. The results of experiment 3, a partial replication of the first two experiments, indicated a general advantage for delayed feedback in terms of errors, time on task, and the percentage of errors that subjects self-corrected. A protocol analysis suggests that immediate feedback competes for working memory resources, forcing out information necessary for operator compilation. In addition, more delayed feedback appears to foster the development of secondary skills such as error detection and self-correction, skills necessary for successful performance once feedback has been withdrawn (Schmidt, Young, Swinnen, & Shapiro, 1989).

Introduction

The timing of feedback is an important issue in skill acquisition. Schmidt, Young, Swinnen, and Shapiro (1989) conducted a study on the timing of feedback with respect to motor skills. They found that shorter feedback latencies improved acquisition and performance while feedback was there, but that delayed feedback resulted in improved subsequent performance once feedback had been withdrawn. Schmidt et al (1989) explain these results in terms of the Guidance Hypothesis. In this view, during the initial stages of skill acquisition immediate feedback guides the behavior of the learner, leading to superior initial performance. This guidance, however, can lead to a dependence on the feedback by obscuring the need to learn secondary skills necessary to perform the task without feedback. The abilities to detect and self-correct errors exemplify such secondary skills.

The results of Lewis and Anderson (1985) are consistent with the predictions of the guidance hypothesis. Within the context of an adventure game, they assessed the effects of immediate and delayed feedback on subsequent performance. Subjects receiving immediate feedback were more likely to select appropriate operators, but those that received delayed feedback were better able to detect errors. Somewhat different results were obtained by Anderson, Conrad, and Corbett (1989). They assessed the effects of immediate and delayed feedback within the context of the GRAPES LISP Tutor. Subjects in the immediate condition moved through the material more quickly than did those in the delay condition, but there were no statistically significant differences in test performance.

This paper will further consider the effects of immediate versus delayed feedback in the context of learning LISP.

Experiment 1

This experiment investigates the effects that various levels of guidance have on complex skill acquisition. Guidance was indirectly manipulated by crossing feedback timing (immediate or delayed) and its focus (directive or nondirective).Directive feedback focuses on the function, or variable, that the student should have used, guiding the subject along a correct solution path. In contrast, nondirective feedback focuses on the function they used incorrectly, but offers no guidance. Immediate feedback provides more guidance than does delayed feedback, by preventing the subject from exploring incorrect solution paths. In this framework, immediate directive feedback provides the most guidance and delayed nondirective the least. These dimensions were integrated into the feedback given to subjects as they solved LISP problems.

Method

Subjects Subjects were members of the Carnegie-Mellon University community with no previous experience with LISP.

Design: The feedback dimensions of timing (immediate or delayed), and focus (directive or nondirective) were crossed, generating 4 cells, with 7 to 9 subjects in each cell. Subjects within a given condition received a single form of feedback throughout the tutoring session. Subjects were matched for math SAT's between the immediate and delayed conditions.

Procedure: Subjects first filled out a background questionnaire about their previous programming experience and their SAT's. Subjects then read materials that familiarized them with both the tutor and the basics of LISP evaluation. Next they solved 4 practice problems involving easily understood arithmetic functions, such as *plus* and *difference*.

Once they had finished the practice problems they were given a second pamphlet describing lists and symbols in LISP and definitions of functions related to their manipulation. After they finished studying the pamphlet, the experimenter started them on the main problems. The solutions to these involved the construction of LISP expressions composed of functions that combine and manipulate lists and symbols. When they made an error, subjects received feedback according to the specifications of their cell.

The general procedure for the second day was similar to the first. However, subjects did not receive any direct feedback from the tutor, though it continued to follow their solutions and record errors. From the subjects' point of view, the tutor acted largely as task master, editor and interpreter; presenting problems and evaluating completed LISP expressions. During both sessions, the subjects received the standard (cryptic) error messages from the LISP interpreter, along with the results of evaluation.

Materials: The subjects learned a modified subset of the LISP extractor and combiner functions (*car, cdr, last, reverse, cons, append & list*). The modifications included changes to the names of some functions such that they were of approximately equal mnemonic value (e.g, *car* to *head*). Other modifications involved restricting the number of arguments that the functions could take. Whereas in standard LISP certain combiner functions, such as *append* and *list*, take a variable number of arguments, our version required that all combiners take two.

The problems were blocked according to difficulty. Within each block, problems were presented randomly. Each of the first seven problems required the use of one of the seven functions and each function was used once within the block. The second block required the use of two functions. Each of the seven was used twice within the block; paired with a different function each time it was used. The third block of problems involved solutions requiring the use of between 3 and 7 functions. In these problems, some functions were used more often than others.

Two problem sets, composed of unique problems, were generated to these specifications. The relative frequency of any given function was comparable in each set.

Tutor Design

Interface: The problem description is displayed in the top portion of the screen (see figure 1). It specifies the initial state, which amounts to the initial variable bindings, and the goal state, the result that the completed expression should return. The subject edits her solution in the space directly below the problem description. Functions and variables that can be used to form the solution are displayed continually. A student selects a solution component, a function or variable, by pressing the key corresponding to the first letter of its name. In figure 1, the student has already entered the function *list* into the editor by pressing the [l]-key. The cursor, here signified by bold brackets, indicates where the component selected next will be entered into the solution. The cursor can be moved to the left and right with the [-]-key and the [+]-key respectively. The editor automatically expands the *list* expression, prompting the student to fill in its first argument, *<expr-1>*. The student's solution continues in figure 2.

==========**Problem Description**==========

Construct and expression such that

if x = (h i j b)
y = (a l m n) then <expression> returns (a b)

(list **[<expr-1>]** <expr-2>)

functions: head tail end reverse list insert append
variables: x y

==========**Help Messages**==========

Figure 1: A screen from the solution to the lhlr problem.

Suppose the student wants to make a list whose first element is the first element of the variable *y*. She accomplishes this by first entering the function *head*, with the [h]-key (step 2). The system expands the *head* expression, and she enters the variable *y* (step 3). The system then prompts her to fill in the second argument of

the *list* expression. Her decision to use the function *tail* (step 4) results in an error.

At this point, since our subject is in the immediate feedback condition, the tutor automatically places the cursor at the point at which the error was detected, and provides feedback (figure 3). She next deletes the offending function (step 5), before successfully completing the problem (step 8). The system then sends the completed expression to the LISP interpreter, and returns the result directly below the subject's expression.

Subject Input	*Result*
	[<function>]
l	(list **[<expr-1>]** <expr-2>)
h	(list (head **[<expr-1>]**) <expr-2>))
y	(list (head y) **[<expr-2>]**)
t	(list (head y) (**[tail]** <expr-1>)) { Feedback }
del	(list (head y) **[<expr-2>]**)
h	(list (head y) (head **[<expr-1>]**))
e	(list (head y) (head (end **[<expr-1>]**))
x	(list (head y) (head (end x))) returns: (a b)

Figure 2: A subjects solution to the lhhr problem, with error.

Directive Feedback

==========Help Messages==========
The system expected to find head at the
current cursor position.

If x = (h i j k) then (head x)
returns: h

Nondirective Feedback

==========Help Messages==========
The system did not expect to find tail at
the current cursor position.

If x = (b g h) then (tail x)
returns: (g h)

Figure 3: Alternate forms of feedback.

Error Detection: The system detects errors by matching the subject's solution against legal solution templates. When it detects a discrepancy between the subject's solution and a legal solution template, it creates a candidate error. Only if all templates fail to match does the system consider an error to be made. When this happens the system must select one of the candidate errors as the basis for generating feedback. It chooses the candidate error associated with the template that most closely matches the student's solution.

Results and Discussion

Training: Two measures of errors were collected. Detected errors are those that the system gives feedback on, or those that cause errors when the final solution is evaluated in LISP. Total errors are all the errors that a subject made; the difference between the two indicates the number errors that were self-corrected. The entire solution cannot always be fully analyzed, since once a function has been marked as an error, it is difficult, if not impossible, to determine whether or not its arguments are in error. The status of these arguments are marked as unknown. The total error count provides a conservative estimate of the total number of errors in the solution. As subjects worked on problems until they arrived at correct solutions, it was possible for them to make multiple errors on each problem.

A two way ANOVA was performed on the subjects' total errors, detected errors, and time on task during training. The factors were feedback timing (immediate or delay) and focus (nondirective or directive). There was a main effect of feedback focus on errors. Subjects receiving directive feedback made fewer detected errors than those receiving nondirective feedback, with means of 22 and 35 errors respectively, $F(1,22)= 6.37$, $p < .02$[1]. Considering that the repair of an error was stated in the directive feedback, this does not come as a surprise. Though subjects receiving nondirective feedback made more errors than did those receiving directive feedback, there was not a significant difference in the time they spent completing the problems, $F(1,22)= .53$, $p < .47$.

Feedback timing had no discernible effect on the total number of errors, $F(1,22) =.09$, $p < .76$, nor on the number of detected errors, $F(1,22)= .24$, $p < .62$. The onset of feedback did, however, have a statistically significant effect on time on task, with subjects in the immediate condition finishing the problems about 40% faster than those working with delayed feedback, $F(1,21) = 13.78$, $p <.002$. This could be an artifact of the editor. Since the arguments to a function are deleted along with a function, a subject receiving delayed feedback might have to retype part of her solution when correcting an error. For example, in order to replace *list* in the expression *(list (head x) y))*

[1]Though the subjects' math SAT scores were used as covariates in the analysis, the unadjusted means are reported.

with append, the subject would have to delete *list*, yielding *<function>*. Next she would have to add *append* and retype the arguments.

Test: As there were no lasting effects of feedback on the second day, this experiment replicates Anderson et al (1989). Subjects in the immediate feedback condition worked through the material in less time than did subjects receiving delayed feedback, yet did not appear to suffer any ill effects.

Experiment 2

There were a number of problems with Experiment 1 that were rectified in Experiment 2. In the delayed condition of Experiment 1 feedback and the results of evaluation were provided automatically as soon as a LISP expression was complete. Once an expression had been filled, the delay condition degenerated into an immediate feedback condition. To prevent this in Experiment 2, feedback and the results of evaluation were provided only after an expression had been filled and the subject hit return as an explicit indication of completing their solution. Though a seemingly minor difference, this gave subjects some control over when feedback was provided, perhaps facilitating self-correction.

The apparent advantage of immediate feedback could have derived from difficulties subjects in the delay condition had with the editor. Moreover, they were only able to self-correct 7% of their errors, before receiving feedback. If, as the guidance hypothesis suggests, one of the principal advantages of delayed feedback results from learning secondary skills such as error detection, such skills should be encouraged by the interface. In Experiment 1 simplicity of use was emphasized in the design of the interface, so that the subjects could focus on learning LISP. Though conceptually simple, it was somewhat cumbersome to use an interface that simultaneously deletes a function and its arguments. Subjects in the delay condition were sometimes forced to delete potentially correct parts of their solution. The interface used in Experiment 2 allowed functions within an expression to be replaced.

Method

Subjects: The thirty-two subjects were comparable to those in Experiment 1.

Design: The feedback dimensions of timing (immediate or delayed), focus (directive or nondirective) were crossed, generating 4 cells, with 8 subjects each. All cells were matched for math sats.

Procedure: From the subjects' point of view the procedure was identical to that of experiment 1.

Results and Discussion

Training: Subjects in the delay condition received feedback on fewer detected errors than did those in the immediate condition, respectively making 17.3 and 36.1 errors, $F(1,23)= 16.2$, $p < .0005$. As in Experiment 1, subjects receiving directive feedback made fewer total (and detected) errors than did subjects receiving nondirective feedback, $F(1,23)= 12.13$, $p < .002$. Subjects in the delayed condition self-corrected 37% of their total errors. By definition subjects in the immediate condition did not have an opportunity to self-correct.

Subjects took longer to finish the problems under nondirective as compared to directive feedback, with respective means of 1824 and 1265 seconds, $F(1,23)= 12.13$, $p < .002$. The advantage for immediate over delayed feedback in experiment 1 was attenuated in this experiment, though it did not disappear completely. Subjects in the immediate condition took less time to complete the problems than did those receiving delayed feedback, taking respectively 1390 and 1699 seconds, $F(1,23)= 2.35$, $p <.13$; subjects in the immediate condition still finished the training problems 18% faster than did those in the delay condition.

Test: Overall subjects in the delay condition made fewer total errors then subjects in the immediate condition with means of 22.0 and 45.3 total errors respectively, $F(1,23)= 9.86$, $p <.005$. Analogous effects were found for time on task. Subjects in the delayed condition spent 1156 seconds on the problems, whereas subjects in the immediate condition took an average of 1727 seconds to complete them, $F(1,23) = 10.35$, $p < .004$. Other dependent measures also showed an advantage of delayed feedback. The total number of steps required to solve the problems, the time required to add a correct step, and the number of errors per correct step were all statistically significant at the .05 level or better.

The guidance hypothesis predicts delayed feedback should foster the development of secondary skills such as self-correction. Consistent with this prediction, subjects in the delay and immediate conditions self-corrected 45% and 30% of their errors respectively, though this difference was not statistically significant.

Experiment 3

The first two experiments yielded somewhat different results. In Experiment 1, the delay and immediate conditions were relatively similar to each other, there were no differences between the delayed (23 total errors & 17% self correction) and immediate (21 & 13%) conditions. In contrast to Experiment 1, where the testing conditions provided little support, the performance of subjects trained in the delay condition of experiment 2 (22 & 45%) was markedly better than that of those trained in the immediate

condition (45 & 30%). These contrasts should be approached with some caution, not only because the comparisons are across experiments, but because of differences in the conditions during testing. We have no indication, for example, of how subjects trained under the delay conditions of Experiment 1 would perform under the testing conditions of Experiment 2.

The primary goal of experiment 3 was to replicate parts of experiments 1 and 2, in particular to determine whether differences in the interface could account for the differing results.

Method

Subjects: The thirty-two subjects were comparable to those in Experiments 1 and 2.

Design: In comparison to the first tutor, the second editor was easier to use and its feedback was more delayed. Though these were the major differences between the tutors used in the first and second experiments, numerous other, seemingly minor, changes were also made. If we were to compare the original conditions of experiment 1 with those of experiment 2 we could only attribute subsequent differences in performance to differences between the tutors as whole. The two versions of the tutor were re-implemented within the same general architecture, so that we can localize the cause of any differences in subjects' performance by systematically manipulating the relevant dimensions.

Subjects were trained in one of four conditions, corresponding to the delayed and immediate conditions of the previous two experiments. They were then tested under either the test conditions of experiment 1 or those of experiment 2. This design generated a total of 8 cells, with 4 subjects per cell. For example, their were 8 subjects trained under the delay conditions of experiment 2. Half of these subjects were tested under the delay conditions of experiment 1 and the other half under delay conditions of experiment 2. As the immediate feedback conditions of experiment 1 and 2 were practically identical, these conditions were collapsed together for the analysis.

Results

Training: The first day results are relatively straightforward. Subjects receiving immediate feedback were somewhat faster than those working in the delayed feedback conditions of either experiment 1 or 2, respectively taking 1713, 2239 and 1795 seconds. $F(1,27)= 2.13$, $p < .14$. Since all subjects received directive feedback, there was relatively little floundering in any of the conditions. Subjects in the immediate condition, and the delay conditions of experiments 1 and 2 made respectively, 21, 24 and 25 errors, $F(1,27)=.5$, $p < .61$.

Test: One subject in the delay condition, who was 2.98 standard deviations slower than the mean, was dropped from the analysis. No other subject was over 2 SD's from the mean on this measure.

Overall, subjects trained in delay conditions (1260 sec) completed the test problems more quickly than did those trained with immediate feedback (1666 sec), $F(1,28)= 2.87$, $p < .10$. Further, subjects in the delay conditions (18 total errors) made fewer mistakes than did subjects in the immediate condition (34 total errors), $F(1,28)= 6.05$, $p < .02$. Subjects in the delay conditions self-corrected 34% of their errors, while subjects in the immediate condition self-corrected only 14% of theirs, $F(1,28)= 6.37$, $p < .015$.

training	testing	time	total errors	self-corrects
Imd	Exp1	1689	33.8	10%
	Exp2	1648	34.5	18%
Exp1	Exp1	1244	18	27%
	Exp2	1462	12	58%
Exp2	Exp1	1166	20.7	30%
	Exp2	1138	22.5	18%

Table 1. Testing results of experiment 3.

As for the replications, subjects trained in the delayed condition of experiment 2 and then tested under the test conditions of experiment 2 performed better than did subjects trained with immediate feedback and tested under these same conditions. Because of the small n, these differences were not significant. The results of the experiment 1 replication were not consistent with those of experiment 1. Subjects trained in the delay condition of experiment 1 performed better than subjects trained with immediate feedback and tested under identical conditions. Both replications suggest an advantage for delayed over immediate feedback.

It could be argued that the differences between the immediate and delayed feedback conditions result solely from the difficulty that subjects trained with immediate feedback have with the interface. To control for this, we can look at subjects' performance on the first step of their solutions, where the advantages that the delay subjects have working with the interface should be diminished. In contrast to the overall result, subjects in the delay (425 sec) condition take longer to make their first moves than do

subjects in the immediate (345 sec) condition, $F(1,28)= 4.55$, $p < .04$. Subjects in the delay (4.3 errors) condition make somewhat fewer mistakes then do those in the immediate (6.3) conditon, $F(1,28)= 2.77$, $p < .11$. This result is consistent with the hypothesis that subjects in the delay condition are spending more time planning their solutions than do subjects who had received immediate feedback during training. In experiments with the geometry tutor, good subjects demonstrated similar behavior (Koedinger & Anderson, 1990).

To summarize Experiment 3, though the subjects in the immediate conditions went through the training problems 18% faster than did those in the delay conditions, they made twice as many errors on the test problems. The results are consistent with experiment 2. This experiment suggests that differences in the interfaces alone do not fully account for the differences in the results between the first and second experiment.

General Discussion

Within the ACT* framework, in order for an appropriate production, or operator, to be compiled, all of the relevant information needs to be in working memory. Whereas Lewis and Anderson (1985) found an advantage for immediate feedback, Anderson et al (1989) did not. Anderson et al (1989) attribute the differences between their results and those of Lewis and Anderson (1985) to differences in the working memory requirements of the two tasks. They suggest that Lewis and Anderson (1985)

> was a situation where the total correct solution was never laid out before subjects and they had to integrate in memory a sequence of moves. In contrast, in the LISP domain students have at the end a working LISP function in front of them to study. (Anderson et 1989)

This interpretation suggests that not only forgetting, but any processing that forces relevant information out of working memory could disrupt production compilation, impairing learning. For example, Sweller (1988) suggests that learning may be hindered by problem solving, since they both rely on the same, limited, cognitive processing capacity. Like problem solving, the processing of feedback could also compete for limited cognitive resources. When a subject is provided with feedback, the feedback necessitates that they set new goals to process it. When they re-emerge from the feedback episode, the previous goals may have been lost, increasing the likelihood that the subject would rely on the feedback. In contrast, if the feedback processing were somewhat less disruptive, then they might return from the feedback episode with their goals intact. A protocol of a subject RM illustrates these costs and some benefits of immediate feedback.

At this point in the protocol, RM's current goal is to get the second element of the list (q a r s), which is bound to the variable y. This can be accomplished by first taking the *tail* (cdr) of the list and then the *head* (car) of the result. Since the LISP interpreter evaluates expressions from the inside out, the arguments to a function are evaluated before the function. This necessitates implementing plans in the reverse order in which they are generated.

In statement 3 he appears to have the components of a successful plan, including the order in which the functions need to be applied. He forgets, however, about the manner in which LISP evaluates its arguments. The system responds with directive interrogative feedback, suggesting the use of head. In statement 4, he proceeds to include *head* in his solution, without understanding why. In statement 5, he wants to manipulate the list, though it would be appropriate to use *tail*, he doesn't think to do this, even though previously it had been a component in his plans; presumably, the plan has been pushed out of working memory. In statement 8 the system has just suggested *tail*, which he does not recognize as part of his original plan. In statement 9, once the components of the solution are largely in place, he reflects on the solution, and learns how to get the second element of a list; he has several more opportunities to do this in the problems, which he does without error.

Rm Protocol

1 **RM:** ..(um)..so I want to manipulate y with some functions. So I'll say ..(ah)..(ah)

2 **E:** What's your reasoning here?

3 **RM:** I'm thinking maybe I would want to use the head function. No the tail function.

edit: (list (**tail** <expr-1>) <expr-2>)
error: The system expected to find head at
the current cursor postion.
What does (head (h i j)) return?

{RM answers question}

4 **RM:** And so that right here. The system expected to find head. I'm not sure why but I will go along with it and say list head.

edit: (list (head **<expr-1>**) <expr-2>)

5 **RM:** And I would want to head the ....oh well maybe I'll say ..(um)..(ah)..I don't want to use reverse no. I don't only want to say y I don't think. (ah) head...

6 E: What are you thinking?

7 RM: (um) Trying to think how I can rearrange the order .. of the y list q a r s without changing a ..without reversing it. Let's see list head ...(um)..(ah)..I would want to say (um)...(hm) I wish I could use something other than head here, but (ah) the system want's it, so its going to get (ah)..(ah)..I guess I could ...(sh).(um).(um). I'll try list again, see if ah.

edit: (list (head (**list** <expr-1> <expr-2>)) <expr-2>)
error: The system expected to find tail at the current cursor position.
What does (tail (w f)) return?

{RM answers question}

8 **RM:** So since it wants tail, I'll give it tail. List head tail and I don't know where the computer is going with this.

edit: (list (head (tail **<expr-1>**)) <expr-2>)

9 **RM:**(um) What are we tailing?. We are tailing y...So that means. Oh I get it, I feel so stupid now. Now I get it because, if you tail q a r s, you get a r s and you head that and you get a which is all I wanted in the first place, so I'll just say y here

edit: (list (head (tail y)) **<expr-2>**)

Figure 4. Verbal Protocol of RM working with immediate feedback.

From this protocol it can be seen that even relatively primitive feedback can provide the basis for quality self-generated explanation (statement 9). Second, that feedback can disrupt working memory, forcing out relevant information (statement 3). Third, it is not difficult to see that if RM had not paused to reflect on his solution, he could have very easily finished the problem, relying on the feedback. R.C. Anderson, Kulhavy & Andre (1972) found that copying feedback such as this impaired learning.

There are clearly advantages to immediate feedback. First, it increases the probability that relevant information will be in working memory (Anderson et al, 1989). Second, it decreases the time spent floundering, focusing the subjects attention on relevant information and decreasing time on task. Along with these benefits, however, are the potential costs. First, subjects can grow dependent on the feedback, so that secondary skills such as error detection and self correction do not develop (Schmidt et al, 1989). Second, immediate feedback may compete for the resources of working memory, impairing learning. Perhaps, a form of feedback can be found in which these benefits and costs are optimally balanced.

References

Anderson, R.C.,Kulhavy, R.W., & Andre, T. (1972). Conditions under which feedback facilitates learning from programmed lessons. *Journal of Educational Psychology*, 63, 186-188.

Anderson, J.R., Conrad, C.G, Corbett, A.T. (in press). Skill Acquisition and the LISP Tutor. *Cognitive Science*,13,467-505.

Koedinger, R., & Anderson, J. R. (1990). Knowledge-Based Environments for Learning and Teaching. Working notes of the AAAI Spring Symposium Series.

Lewis, M.W., & Anderson, J.R. (1985). Discrimination of Operator Schemata in Problem Solving: Learning from Examples. *Cognitive Psychology*, **17**, 26-65.

Schmidt, R.A., Young, D.E., Swinnen, S., & Shapiro, D.C. (1989). Summary Knowledge of Results for Skill Acquisition: Support for the Guidance Hypothesis. *Journal of Experimental Psychology: Learning, Memory, and Cognition*,**15**(2), 352-359.

Sweller, J. (1988). Cognitive Load During Problem Solving: Effects on Learning. *Cognitive Science*,12,257-285.

LEARNING THE STRUCTURE OF EVENT SEQUENCES

Axel Cleeremans
James L. McClelland

Department of Psychology
Carnegie Mellon University

ABSTRACT

How is complex sequential material acquired, processed, and represented when there is no intention to learn ? Recent research (Lewicki, Hill & Bizot, 1988) has demonstrated that subjects placed in a choice reaction time task progressively become sensitive to the sequential structure of the stimulus material despite their unawareness of its existence. This paper aims to provide a detailed information-processing model of this phenomenon in an experimental situation involving complex and probabilistic temporal contingencies. We report on two experiments exploring a 6-choice serial reaction time task. Unbeknownst to subjects, successive stimuli followed a sequence derived from "noisy" finite-state grammars. After considerable practice (60,000 exposures), subjects acquired a body of procedural knowledge about the sequential structure of the material, although they were unaware of the manipulation, and displayed little or no verbalizable knowledge about it. Experiment 2 attempted to identify limits on subjects' ability to encode the temporal context by using more distant contingencies that spanned irrelevant material. Taken together, the results indicate that subjects become progressively more sensitive to the temporal context set by previous elements of the sequence, up to three elements. Responses are also affected by carry-over effects from recent trials. A PDP model that incorporates sensitivity to the sequential structure and carry-over effects is shown to capture key aspects of both acquisition and processing of the material.

INTRODUCTION

In many situations, learning does not proceed in the explicit and goal-directed way characteristic of traditional models of cognition (Newell & Simon, 1972). Rather, it appears that some of our knowledge and skills are acquired in an incidental and unintentional manner. Indeed, many studies (see Reber, 1989; for a review) have documented dissociations between task performance and reportable knowledge. The classic result in these experimental situations is that "subjects are able to acquire specific procedural knowledge (i.e. processing rules) not only without being able to articulate what they have learned, but even without being aware that they had learned anything" (Lewicki, Czyzewska & Hoffman, 1987).

Although controversy still pervades the field, at least two different implicit learning paradigms have yielded consistent and robust results : artificial language learning (Reber, 1967, 1989), and sequential pattern acquisition (Nissen & Bullemer, 1987; Lewicki, Czyzewska & Hoffman, 1987; Lewicki, Hill & Bizot, 1988; Willingham, Nissen & Bullemer, 1989; Cohen, Ivry & Keele, 1990). Related research with neurologically impaired patients also provides strong evidence for the existence of learning processes that do not entail or require awareness of the results or of the learning experience itself (see Schacter, 1987, for a review).

Despite this wealth of evidence documenting implicit learning phenomena, few models of the mechanisms involved have been proposed. This lack of formalization can doubtless be attributed to the difficulty of assessing subject's knowledge when it does not lend itself easily to verbalization.

Nevertheless, an important first step in the direction of understanding the relationship between awareness, attention and learning consists of attempting to identify mechanisms that account for subjects's performance in implicit learning situations. In the present paper, we report on a series of experiments inspired by Lewicki, Hill & Bizot's (1988) paradigm, and propose a detailed information processing model of the task. These experiments placed subjects in a choice reaction time task, and manipulated the sequential contingencies of the material in a novel way that allows detailed data about subject's representations of the temporal structure to be obtained.

The main results of our experiments indicate that subjects unintentionally acquire a complex body of knowledge about the temporal structure of the material. We describe a PDP model that implements a proposed mechanism to account for performance in this task. The model -- trained in exactly the same conditions as subjects -- captures key aspects of both acquisition and performance in this task. Its core mechanism implements the hypothesis that sequential structure gets induced as a direct result of an encoding of events *together* with an internal representation of the temporal context.

EXPERIMENT 1

Subjects were exposed to a six-choice reaction time task. The entire experiment was divided in 20 sessions. Each session consisted of 20 blocks of 150

This research was supported by a grant from the National Fund for Scientific Research (Belgium) to the first author and by an NIMH RSDA to the second author. Address correspondence to : Axel Cleeremans, Department of Psychology, Carnegie Mellon University, Pittsburgh PA 15213: or email to *cleeremans@psy.cmu.edu.*

trials. On any of the 60,000 trials, a stimulus could appear at one of six positions arranged in a horizontal line on a computer screen. The task consisted of pressing as fast and as accurately as possible on one of six corresponding keys. Unbeknownst to subjects, the sequential structure of the stimulus material was manipulated. Stimuli were generated using a small finite-state grammar that defined legal transitions between successive trials. Some of the stimuli, however, were not "grammatical" . Indeed, on each trial, there was a 15% chance of substituting a random stimulus to the one prescribed by the grammar. This "noise" served two purposes. First, it ensured that subjects could not simply memorize the sequence of stimuli, and hindered their ability of detecting regularities in an explicit way. Second, since each stimulus was possible on every trial (if only in a small proportion of the trials), we could obtain detailed information about what stimuli subjects did or did not expect at each step.

If subjects become increasingly sensitive to the sequential structure of the material over training, one would thus predict an increasingly large difference in the reaction times elicited by predictable and unpredictable stimuli. Further, detailed analyses of the RTs to particular stimuli in different temporal contexts should reveal differences that reflect subject's encoding of the sequential structure of the material.

Method

Subjects. Six subjects (CMU staff and students) aged 17-42 participated in the experiment. Each subject was paid $100 for his participation in the 20 sessions of the experiment, and received a bonus of up to $50 based on speed and accuracy.

Apparatus and display. The experiment was run on a Macintosh II computer. The display consisted of six dots arranged in a horizontal line on the computer's screen. Each screen position was paired with a key on the computer's keyboard, also arranged in a line ('Z', 'X', 'C', 'B', 'N', 'M'). The stimulus was a small black circle that appeared immediately below one of the six dots. The timer was started at the onset of the stimulus and stopped by the subject's response. The RSI was 120 msec.

Procedure. Subjects received detailed instructions during the first meeting. They were told that the purpose of the experiment was to "learn more about the effect of practice on motor performance". Both speed and accuracy were stressed as being important. After receiving the instructions, subjects were given 3 practice blocks of 15 random trials each at the task. A schedule for the 20 experimental sessions was then elaborated. Most subjects followed a regular schedule of two sessions a day.

Stimulus Material. Stimuli were generated on the basis of the small finite-state grammar shown in Figure 1. Finite-State grammars consist of nodes connected by labeled arcs. Expressions of the language are generated by starting at node #0, choosing an arc, recording its label, and repeating this process with the next node. The vocabulary associated with the grammar we used consists of six letters ('T', 'S', 'X', 'V', 'P', and 'Q'), each represented twice on different arcs of the grammar (as denoted by the subscript of each letter). This results in highly context-dependent transitions, as identical letters can be followed by different sets of successors as a function of their position in the grammar (For instance, 'S_1' can only be followed by 'Q', but 'S_2' can be followed by either 'V' or 'P'). The grammar was constructed so as to avoid direct repetitions of a particular letter, since it is known (Hyman, 1953; Bertelson, 1961) that repeated stimuli elicit shorter reaction times independently of their probability of presentation. Finally, note that the grammar loops onto itself: the first and last nodes, both denoted by the digit 0, are actually the same.

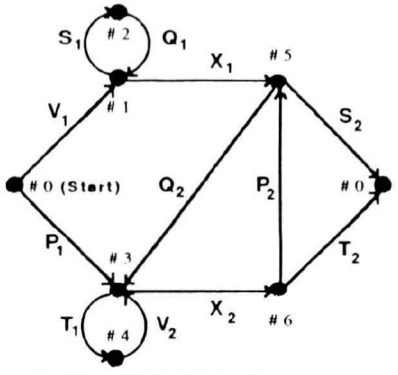


Figure 1 : The Finite State Grammar used to generate the stimulus sequence in Experiment 1.

Stimulus generation proceeded as follows. On each trial, three steps were executed in sequence. First, an arc was selected at random among the possible arcs coming out of the current node, and its corresponding letter recorded. The current node was set to be node #0 on the very first trial of any block, and was updated on each trial to be the node pointed to by the selected arc. Second, in 15 % of the cases, another letter was substituted to the letter recorded at step 1 by choosing it at random among the five remaining letters in the grammar. Third, the selected letter was used to determine the screen position at which the stimulus would appear. A 6 x 6 Latin Square design was used, so that each letter corresponded to each screen position for exactly one of the six subjects.

Design. The experiment consisted of 20 sessions of 20 blocks of 155 trials each. Each block was initiated by a "Get ready" message and a warning beep. After a short delay, 155 trials were presented to the subject. The first five trials of each block were entirely random so as to eliminate initial variability in the responses. These data points were not recorded. The next 150 trials were generated according to the procedure described above. After each block, the computer paused for approximately 30 seconds. The message "Rest Break" was displayed on the screen, along with information about subjects's performance (mean RT and accuracy for the last block, and amount earned).

Results & Discussion

Figure 2 shows the average RTs on correct responses for each of the 20 experimental sessions, plotted separately for predictable and unpredictable trials. A general practice effect is readily apparent, as well as an increasingly large difference between predictable and unpredictable trials. A two-way ANOVA with repeated measures on both factors (practice [20 levels] X trial type [grammatical vs.

ungrammatical]) revealed significant main effects of practice, $F(19,95) = 9.491$, $p < .001$; and of trial type, $F(1,5) = 105.293$, $p < .001$; as well as a significant interaction, $F(19,95) = 3.022$, $p < .001$. It appears that subjects become increasingly sensitive to the sequential structure of the material. Yet, when interviewed after the task, all subjects reported feeling that the sequence was random, and failed to report noticing any pattern in the data but small alternations (e.g. the loops on nodes #2 and #4).

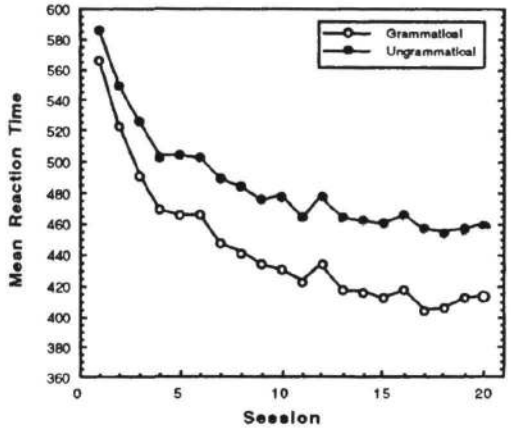


Figure 2 : Mean RTs for grammatical and ungrammatical trials for each of the 20 sessions of Experiment 1.

Accuracy averaged 98.12% over all trials. The small difference between accuracy on predictable (98.35%) and unpredictable (96.68%) trials was not significant.

One mechanism that would account for the progressive differentiation between predictable and unpredictable trials consists of assuming that subjects, in attempting to optimize their reaction times, progressively come to anticipate successive events on the basis of an increasingly large temporal context set by previous elements of the sequence. In the grammar we used, most elements can be perfectly anticipated on the basis of two elements of temporal context, but some of them require three or even four elements of temporal context to be maximally disambiguated. For instance, the path 'SQ' (leading to node #2) occurs only once in the grammar and can only be legally followed by 'S' or by 'X'. In contrast, the path 'TVX' can lead to either node #5 or node #6, and is therefore not sufficient to perfectly distinguish between stimuli that occur only at node #5 ('S' or 'Q') and stimuli that occur only at node #6 ('T' or 'P'). One would assume that subjects initially respond to the predictions entailed by the shortest paths, and progressively become sensitive to the higher-order contingencies as they encode more and more temporal context.

A simple analysis that would reveal whether or not subjects are indeed basing their performance on an encoding of an increasingly large temporal context was conducted. Its general principle consists of comparing the data with the probability of occurrence of the stimuli *given* different amounts of temporal context.

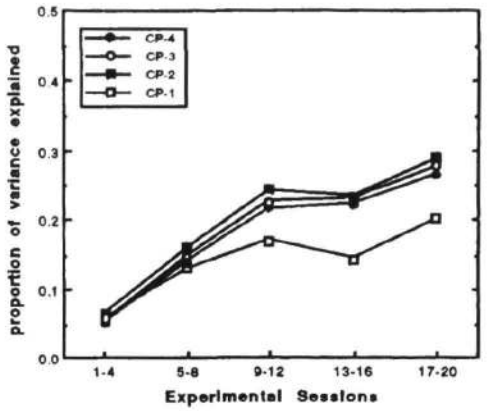


Figure 3 : Correspondence between the human responses and CPs after paths of length 1-4 during successive blocks of four simulated sessions.

First, we estimated the conditional probabilities (CPs) of observing each letter as the successor of every grammatical path of length 1, 2, 3 and 4 respectively. Next, the average RT for each successor to paths of length 4 were computed, separately for successive blocks of four experimental sessions. Finally, 20 separate regression analyses were conducted, using each of the four sets of CPs as predictor, and each of the five sets of mean RTs as dependent variable. If subjects are encoding increasingly large amounts of temporal context, we would expect the variance in the distribution of their responses at successive points in training to be better explained by CPs of increasingly higher statistical orders.

Figure 3 illustrates the results of these analyses. Each point on the figure represents the r-squared coefficient of a specific regression analysis. Points corresponding to analyses conducted with the same amount of temporal context (1 - 4 elements) are linked together. Although the overall fit is rather low (Note that the vertical axis only extends to 0.5), the figure nevertheless reveals that subjects become increasingly sensitive to the temporal context set by previous elements of the sequence. One can see that the correspondence with the first-order CPs tends to level off below the fits for the second, third and fourth orders. The fits to the second, third and fourth order paths are highly similar in part because their associated CPs are themselves highly similar.

In order to assess more directly whether subjects are able to encode three or four letters of temporal context, several analyses on specific successors of specific paths were conducted. One such analysis involved several paths of length 3. These paths were the same in their last two elements, but differed in their first element as well as in their legal successors. For

example, we compared 'XTV' versus 'PTV' and 'QTV', and examined RTs for the letters 'S' (legal only after 'XTV') and 'T' (legal only after 'PTV' or 'QTV'). If subjects are sensitive to three letters of context, their response to an 'S' should be relatively faster after 'XTV' than in the other cases, and their response to a 'T' should be relatively faster after 'PTV' or 'QTV' than after 'XTV'. Averaging over all candidate contexts of this type, we found that a slight advantage for the legal successors emerged in sessions 8-12 and remained present over sessions 13-16 and 17-20 ($p < .05$). Thus there appears to be evidence of sensitivity to at least three elements of temporal context. However, no sensitivity to the first element of otherwise identical paths of length 4 (e.g. 'XTVX' vs. 'PTVX and 'QTVX') was found, even during sessions 17-20.

EXPERIMENT 2

Experiment 1 demonstrated that subjects progressively become sensitive to the sequential structure of the material and seem to be able to maintain information about the temporal context for up to three steps. The temporal contingencies characterizing this grammar were relatively simple, however, since in most cases, only two elements of temporal context are needed to disambiguate the next event perfectly.

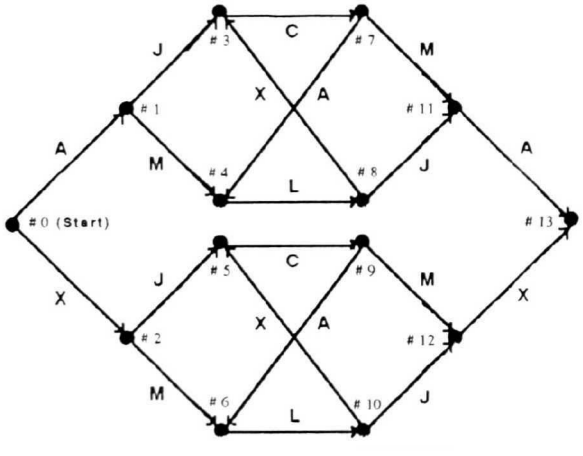


Figure 4 : The Finite State Grammar used to generate the stimulus sequence in Experiment 2.

Further, contrasting long-distance dependencies were not controlled for their overall frequency. In Experiment 2, a more complex grammar (Figure 4) was used in an attempt to identify limits on subjects' ability to maintain information about more distant elements of the sequence. In this grammar, the last element ('A' or 'X') is contingent on the first one (also 'A' or 'X'). Information about the first element, however, has to be maintained across either of the two embeddings in the grammar, and is totally irrelevant for predicting the elements of the embeddings. Further, the two embeddings are identical. Thus, in order to accurately predict the last element at nodes #11 or #12, one needs to maintain information across a minimum of three intervening steps. Accurate expectations about the nature of the last element would be revealed by a difference in the RT elicited by the letters 'A' and 'X' at nodes #11 and #12 ('A' should be faster than 'X' at node #11, and vice-versa). Naturally, there was again a 15% chance of substituting another letter to the one prescribed by the grammar. Further, in order to avoid direct repetitions between the letters that precede and follow node #13, a small loop was inserted at node #13. One random letter was always presented at this point; after which there was a 40% chance of staying in the loop on subsequent steps.

Method

Six new subjects (CMU undergraduates and graduates, aged 19-35) participated in Experiment 2. The design of Experiment 2 was otherwise identical to that of Experiment 1.

Results & Discussion

Figure 5 shows the main results of Experiment 2. They closely replicate the general results of Experiment 1, although subjects were a little bit faster overall in Experiment 2. A two-way ANOVA with repeated measures on both factors (practice [20 levels] X trial type [grammatical vs. ungrammatical]) again revealed significant main effects of practice, $F(19,95) = 32.011$, $p < .001$; and of trial type, $F(1,5) = 253.813$, $p < .001$; as well as a significant interaction, $F(19,95) = 4.670$, $p < .001$.

Accuracy was 97.00% over all trials. The difference between grammatical (97.60%) and ungrammatical (95.40%) was significant; $t(5) = 2.294$, $p < 0.5$.

Of greater interest are the results of analyses conducted on the responses elicited by the successors of the four shortest paths starting at node #0 and leading to either node #11 or node #12 ('AJCM', 'AMLJ', 'XJCM' & 'XMLJ'). Among those paths, those beginning with 'A' predict 'A' as their only possible successor, and vice-versa for paths starting with 'X'. This only holds, though, if all four letters of each path are encoded. Indeed, the sub-paths 'JCM' and 'MLJ' undifferentially predict 'A' or 'X' as their possible successors. The RTs on legal successors of each of these four paths (i.e. 'A' for 'AJCM' and 'AMLJ'; and 'X' for 'XJCM' and 'XMLJ') were averaged together and compared to the average RT on their illegal successors (i.e. 'X' for 'AJCM' and 'AMLJ'; and 'A' for 'XJCM' and 'XMLJ'), thus yielding two scores. Any significant difference between these two scores would mean that subjects are discriminating between legal and illegal successors of these four paths, thereby

suggesting that they have been able to maintain information about the first letter of each path over three irrelevant steps. The mean RT on legal successors over the last four sessions of the experiment was 384.896, and the corresponding score for illegal successors was 387.847. A paired t-test on this difference failed to reach significance ($t(5) = 0.571$, $p > 0.05$).

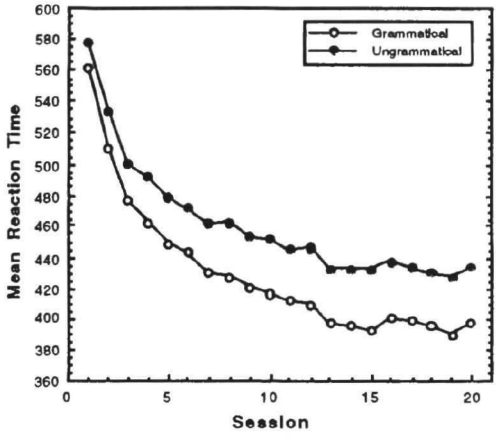


Figure 5 : Mean RTs for grammatical and ungrammatical trials for each of the 20 sessions of Experiment 2.

To sum up, subjects do not appear to be able to encode long-distance dependencies when they span 3 items of embedded independent material; at least, they cannot do so in the amount of practice used here. However, there is clear evidence of sensitivity to at least the previous two elements of temporal context.

SIMPLE RECURRENT NETWORKS

Early models of sequence processing (e.g. Estes' "statistical learning theory") have typically assumed that subjects somehow compute the conditional probabilities for all relevant statistical orders, but failed to show *how* subjects might come to represent or compute them. In the following, we present a model of sequence processing that comes to elaborate its own internal representations of the temporal context despite very limited processing resources. The model consists of a Simple Recurrent back-propagation Network ('SRN', see Elman, 1988; Cleeremans, Servan-Schreiber & McClelland, 1989).

In the SRN (Figure 6), the hidden unit layer is allowed to feed back on itself, so that the intermediate results of processing at time **t-1** can influence the intermediate results of processing at time **t**. In practice, the SRN is implemented by copying the pattern of activation on the hidden units onto a set of "context units" which feed into the hidden layer, along with the input units. All the forward-going connections in this architecture are modified by back-propagation. The recurrent connections from the hidden layer to the context layer implement a simple copy operation and are *not* subject to training.

As reported elsewhere (Cleeremans, Servan-Schreiber & McClelland, 1989), we have shown that an SRN trained to *predict* the successor of each element of a sequence presented one element at a time can learn to perform this "prediction task" perfectly on simple finite-state grammars like the one used in Experiment 1. Following training, the network produces the conditional probabilities of presentation of all possible successors of the sequence. Since all letters of the grammar were inherently ambiguous (i.e. predicting them requires more than the immediate predecessor to be encoded), the network must have developed representations of entire subsequences of events. Note that the network is never presented with more than one element of the sequence at a time. Thus, it has to elaborate its own internal representations of as much temporal context as is needed to achieve optimal predictions.

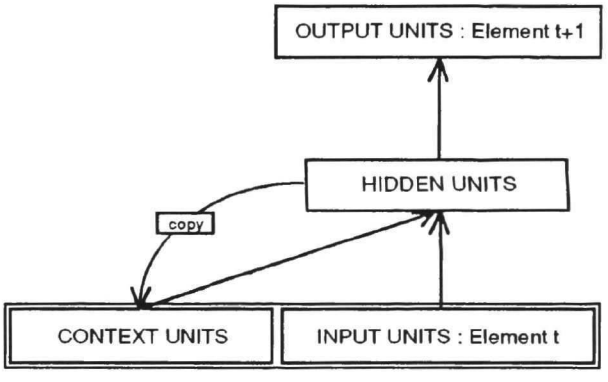


Figure 6 : The simple recurrent network.

A complete analysis of the learning process is too long to be presented here (a full account is given in Servan-Schreiber, Cleeremans & McClelland, 1988), but the key points are as follows : As the initial papers about back-propagation (e.g. Rumelhart, Hinton & Williams, 1986) pointed out, the hidden unit patterns of activation represent an "encoding" of the features of the input patterns that are relevant to the task. In the SRN, the hidden layer is presented with information about the current letter, but also -- on the context layer -- with an encoding of the relevant features of the previous letter. Thus, a given hidden layer pattern can come to encode information about the relevant features of two consecutive letters. When this pattern is fed back on the context layer, the new pattern of activation over the hidden units can come to encode information about three consecutive letters, and so on. In this manner, the context layer patterns can allow the network to learn to maintain prediction-relevant features of an entire sequence.

To model our experimental situation, we used an

SRN with 15 hidden units and local representations on both the input and output pools (i.e. each unit corresponded to one of the 6 stimuli). The network was trained to *predict* each element of a continuous sequence of stimuli generated in exactly the same conditions as for the human subjects. On each step, a letter was generated from the grammar as described above, and presented to the network by setting the activation of the corresponding input unit to 1.0. Activation was then allowed to spread to the other units of the network, and the error between its response and the *actual successor* of the current stimulus was then used to modify the weights.

During training, the activation of each output unit was recorded on every trial and transformed into Luce ratios to normalize the responses. For the purpose of comparing the model's and the subject's responses, we assumed 1) that the normalized activations of the output units represent response tendencies, and 2) that there is a linear reduction in RT proportional to the relative strength of the unit corresponding to the correct response[2].

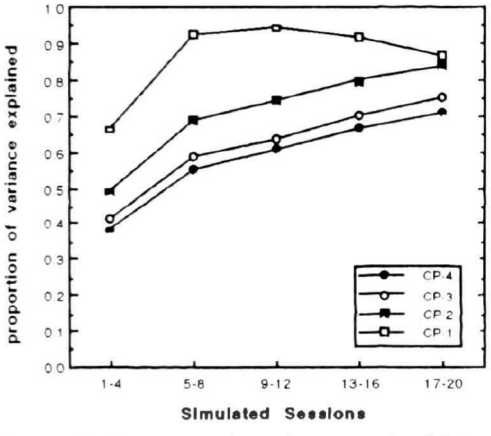


Figure 7 : Correspondence between the SRN's responses and CPs after paths of length 1-4 during successive blocks of four simulated sessions.

This data was first analyzed in the same way as for Experiment 1 subjects, and compared to the CPs of increasingly higher statistical orders in 20 separate regression analyses. The results are illustrated in Figure 7. In stark contrast with the human data (Figure 3, note the scale difference), the variability in the model's responses appears to be very strongly determined by the probabilities of particular successor letters given the temporal context. The figure also reveals that the model's behavior is dominated by the first-order CPs for most of the training, but that it becomes progressively more sensitive to the second and higher order CPs. If training was to be continued beyond 60,000 exposures, the model's responses would come to approximate increasingly higher CPs.

Figure 8 illustrates a more direct comparison between the model's responses at successive points in training with the corresponding human data. First, we computed the average RT of each letter *at each node of the grammar*. This yields a set of 42 data points (Due to noise, each of the six letters may occur at any of the seven different nodes). This analysis was conducted on RTs averaged over blocks of four successive experimental sessions, thus yielding five different sets of data. Next, a similar analysis was conducted on the model's responses. Finally, we conducted 25 separate regression analyses on these data. Each point in Figure 8 represents the r-squared coefficient of a regression analysis using the model's responses at a particular point in training as predictor and the human data as dependent variable. One would expect the model's early performance to be a better predictor of the subjects's early behavior, and vice-versa for later points in training.

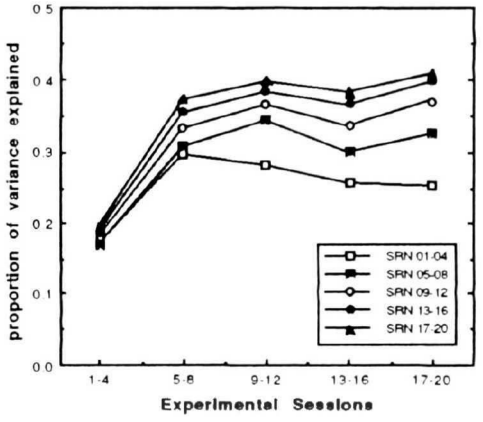


Figure 8 : Correspondence between the SRN's responses and the corresponding human data during successive blocks of four sessions of training.

It is obvious that the model is not very good at capturing subjects's behavior : the overall fit is relatively low (note that the vertical axis only goes up to 0.5), and reflects only weakly the expected progressions. Basically, too much of the variance in the model's performance is accounted for by sensitivity to the temporal context.

However, exploratory examination of the data revealed that performance in this task depends on three other factors (in addition to the conditional probability of appearance of a stimulus) :

First of all, it appears that a response that is actually executed remains primed for a number of subsequent trials (Remington, 1969). If it follows itself immediately, there is about 60 to 90 msec of facilitation, depending on other factors. If it follows after a single intervening response (as in 'VT-V', for example), there is about 25 msec of facilitation if the letter is grammatical at the second occurrence, and 45 msec if it is ungrammatical.

[2] Naturally, the second assumption is a simplification. We are currently in the process of exploring more realistic versions of this assumption.

The second factor may be related: responses that are grammatical at trial **t** but do not actually occur remain primed at trial **t+1**; the effect is somewhat weaker, averaging about 30 msec. The first two factors may be summarized by assuming that activations at time **t** decay gradually over subsequent trials, and responses that are actually executed become fully activated, while those that are not executed are only partially primed.

The third factor is a priming, not of a particular response, but of a particular sequential pairing of responses. This can best be illustrated by a contrasting example, in which the response to the second 'X' is compared in 'QXQ-X' and 'VXQ-X'. The response to the second X tends to be about 10 msec faster in cases like 'QXQ-X', where the 'X' follows the same predecessor twice in a row, than it is in cases like 'VXQ-X', in which the first 'X' follows one letter and the second follows a different letter.

This third factor can perhaps be accounted for in several ways. We have explored the possibility that it results from a rapidly decaying component to the increment to the connection weights mediating the associative activation of a letter by its predecessor. Such "fast" weights have been proposed by a number of investigators (McClelland & Rumelhart, 1985; Hinton & Plaut, 1987). The idea is that when 'X' follows 'Q', the connection weights underlying the prediction that 'X' will follow 'Q' receives an increment which has a short-term component in addition to the standard long-term component. This short-term increment is still present in sufficient force to influence the response to a subsequent 'X' that follows a immediately subsequent 'Q'.

In light of these analyses, one possibility for the relative failure of the original model to account for the data is that the SRN is partially correct, but that human responses are also affected by rapidly decaying activations and adjustments to connection weights from preceding trials. To test this idea, we incorporated both kinds of mechanisms into a second model.

This new simulation model was exactly the same as before, except for the following two changes :

First, it was assumed that pre-activation of a particular response was based, not only on activation coming from the network but also on a decaying trace of the previous activation:

$$respact[i](\mathbf{t}) = act[i](\mathbf{t}) + (1 - act[i](\mathbf{t})) * \mathbf{k} * respact[i](\mathbf{t} - \mathbf{1})$$

where act(**t**) is the activation of the unit based on the network at time **t**, and respact(**t**) is a kind of non-linear running average that remains bounded between 0 and 1. When a particular response is executed, the corresponding respact is set to 1.0. The constant **k** is set to 0.5, so that the half-life of a response activation is one time step.

The second change is simply to assume that when weights are changed by the back-propagation learning procedure, there are two components, one of which is a small (epsilon = 0.15) but effectively permanent change (i.e., a decay rate slow enough to ignore for present purposes) and the other of which is a larger (epsilon = 0.2) change that has a half-life of a single time-step.

With these changes in place, we observed that, of course, the proportion of the variance in the model accounted for by predictions based on one to four letters of temporal context is dramatically reduced (Figure 9). More interestingly, the pattern of change in these measures, as well as the overall fit, is now quite similar to that seen in the data (Figure 3).

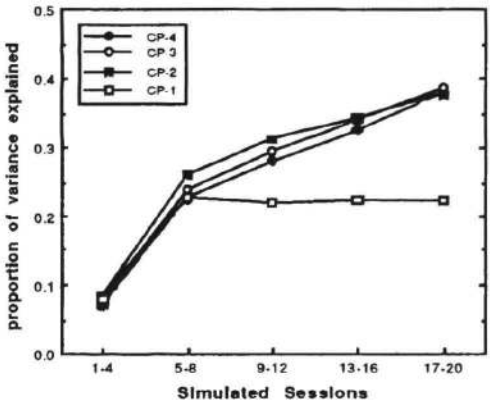


Figure 9 : Correspondence between the augmented SRN's responses and CPs after paths of length 1-4 during successive blocks of four simulated sessions.

Indeed, there is a similar progressive increase in the correspondence with the higher-order CPs, with the curve for the first-order CPs leveling off relatively early, as in the human data.

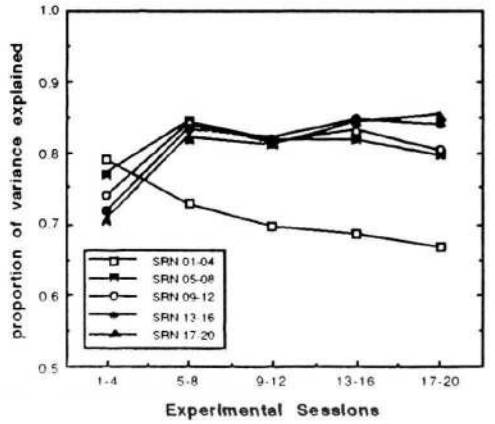


Figure 10 : Correspondence between the augmented SRN's responses and the corresponding human data during successive blocks of four sessions of training.

A more direct indication of the good fit provided by the current version of the model is given by the fact

that it now correlates extremely well with the performance of the subjects (Figure 10; compare with the same analysis illustrated in Figure 8). Late in training, the model explains about 86% of the variance of the corresponding human data. Close inspection of the figure also reveals, that, as expected, the SRN's early distribution of responses is a better predictor of the corresponding early human data. This correspondence gets inverted later on, thereby suggesting that the model now captures key aspects of acquisition as well. Indeed, at every point, the best prediction of the human data is the simulation of the corresponding point in training.

GENERAL DISCUSSION

In Experiment 1, subjects were exposed to a 6-choice serial reaction time task for 60,000 trials. The sequential structure of the material was manipulated by generating successive stimuli on the basis of a small finite-state grammar. On some of the trials, random stimuli were substituted to those prescribed by the grammar. The results clearly support the idea that subjects become increasingly sensitive to the sequential structure of the material. Indeed, the smooth differentiation between predictable and unpredictable trials can only be explained by assuming that the temporal context set by previous elements of the sequence facilitates or interferes with the processing of the current event. Experiment 2 showed that subjects were relatively unable to maintain information about long-distance contingencies that span irrelevant material.

Taken together, these results suggest that in this task, subjects gradually acquire a complex body of procedural knowledge about the sequential structure of the material. They are clearly sensitive to more than just the immediate predecessor of the letter; indeed, there is evidence of sensitivity to differential predictions based on two and even three elements of context. However, sensitivity to temporal context is clearly limited: even after 60,000 trials of practice, there is no evidence of sensitivity to fourth-order temporal context. Of course, it remains possible that the subjects would eventually discover the fourth-order structure, just as the model can do.

The augmented SRN model provides a detailed, mechanistic , and fairly good account of the data. At this point it is difficult to be certain whether the model is capable of offering a complete account of all of the structure in the data. First, we have not explored the parameter space very extensively to discover whether it is possible to improve on the existing fit; and second, it is not clear just how much more systematic (as opposed to random) variance there is in the data to be accounted for.

It is often claimed that learning can proceed without explicit awareness (e.g. Reber, 1989; Willingham, Nissen & Bullemer, 1989). In our case, it appears that subjects do become aware of the alternations that occur in the grammar (e.g 'SQSQ' and 'VTVT' in Experiment 1), but have little reportable knowledge of any other contingencies. Given the fairly close correspondence of the augmented SRN with the subjects's performance, this class of model would appear to offer a viable framework for modeling this type of implicit learning.

REFERENCES

Bertelson, P. (1961). Sequential redundancy and speed in a serial two-choice responding task. *QJEP*, **13**, 90-102.

Cleeremans, A., Servan-Schreiber, D., & McClelland, J.L. (1989). Finite state automata and simple recurrent networks. *Neural Computation*, **1**, 372-381.

Cohen, A. Ivry, R.I., & Keele, S.W. (1990). Attention and structure in sequence learning. *JEP : LMC*, **16**, 17-30.

Elman, J.L. (1988). *Finding structure in time*. CRL Technical report 8801, Center for research in language, University of California, San Diego.

Hinton, G.E., & Plaut, D.C. (1987). Using fast weights to deblur old memories. In *Proc. 9th. Int. Conf. Cog. Sci. Society*.

Hyman, R. (1953). Stimulus information as a determinant of reaction time. *JEP*, **45**, 188-196.

Lewicki, P., Czyzewska, M., & Hoffman, H. (1987). Unconscious acquisition of complex procedural knowledge. *JEP : LMC*, **13**, 523-530.

Lewicki, P., Hill, T., & E. Bizot, E. (1988). Acquisition of procedural knowledge about a pattern of stimuli that cannot be articulated. *Cognitive Psychology*, **20**, 24-37.

McClelland, J.L., & Rumelhart, D.E. (1985). Distributed memory and the representation of general and specific information. *JEP : General*, **114**, 159-188.

Newell, A., & Simon, H.A. (1972). *Human problem solving*. Englewood Cliffs, N.J. : Prentice-Hall.

Nissen, M. J., & Bullemer, P. (1987). Attentional requirements of learning : Evidence from performance measures. *Cognitive Psychology*, **19**, 1-32.

Reber, A.S. (1967). Implicit learning of artificial grammars. *JVLVB*, **6**, 855-863.

Reber, A.S. (1989). Implicit learning and tacit knowledge. *JEP : General*, **118**, 219-235.

Remington, R.J. (1969). Analysis of sequential effects in choice reaction times. *JEP*, **82**, 250-257.

Rumelhart, D.E., Hinton, G., & Willians, R.J. (1986). Learning internal representations by error propagation. In Rumelhart, D.E. and McClelland, J.L. (Eds.), *Parallel Distributed Processing, I : Foundations*. Cambridge, MA : MIT Press.

Schacter, D.L. (1987). Implicit memory : History and current status. *JEP : LMC*, **13**, 501-518.

Servan-Schreiber, D., Cleeremans, A., & McClelland, J.L. (1988). *Encoding sequential structure in simple recurrent networks*, Technical Report CMU-CS-88-183, Department of Computer Science, Carnegie Mellon University.

Willingham, D.B., Nissen, M.J., & Bullemer, P. (1989). On the development of procedural knowledge. *JEP : LMC*, **15**, 1047-1060.

Explanation-based learning of correctness: Towards a model of the self-explanation effect

Kurt VanLehn, William Ball, Bernadette Kowalski
Depts. of Computer Science and Psychology
Carnegie-Mellon University

Abstract

Two major techniques in machine learning, *explanation-based learning* and *explanation completion*, are both superficially plausible models for Chi's self-explanation effect, wherein the amount of explanation given to examples while studying them correlates with the amount the subject learns from them. We attempted to simulate Chi's protocol data with the simpler of the two learning processes, explanation completion, in order to find out how much of the self-explanation effect it could account for. Although explanation completion did not turn out to be a good model of the data, we discovered a new learning technique, called *explanation-based learning of correctness*, that combines explanation-based learning and explanation completion and does a much better job of explaining the protocol data. The new learning process is based on the assumption that subjects use a certain kind of plausible reasoning.

Chi, Bassok, Lewis, Reimann and Glaser (1989) showed that self-explanation correlates with better learning. They analyzed protocols of 8 students who learned physics from a standard college textbook. The students first studied prerequisite material and took tests to show that they had learned it. They next read a textbook chapter on Newtonian dynamics, then studied examples of worked-out physics problems. Lastly, they solved problems on their own. All this work was done without feedback from an instructor. Protocols were taken during the example-studying phase and during the problem-solving phase. The percentage of problems answered correctly during the problem-solving phase was used to divide students into 4 Good students and 4 Poor students. As both groups of students performed equally well on pre-tests, it appears that the Good students learned more from the examples than the Poor students. In examining the causes of this, Chi et al. found two consistent behavioral differences between the groups. (1) The Poor students studied the examples by merely reading and paraphrasing them, in contrast to the the Good students, who explained each line of the example to themselves in considerable detail. (2) Although all students would utter sporadic comments indicating whether they understood an example's line, the Poor students tended to say that they *did* understand the line, while the Good students tended to say that they *did not* understand the line. Both correlations were large and statistically significant. The results have been replicated twice (Pirolli & Bielaczyc, 1989; Ferguson-Hessler & de Jong, 1990). Chi et al. labeled the Good student's example-studying process *self-explanation*. So the basic finding is that self-explanation correlates with better learning.

The research reported here is aimed at explaining the self-explanation finding by building computer simulations of the processes used by Good and Poor students and comparing the simulations' behavior with Chi's protocol data. We report here on the first version of the simulation program, Cascade1, indicate its strengths and weaknesses, and sketch the idea behind the next version, Cascade2. Before describing the program per se, it is worth reviewing some recent work in machine learning that appears to offer an account of the self-explanation effect.

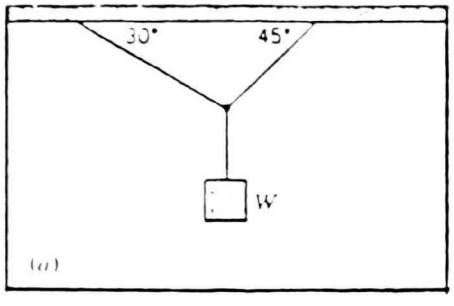
Explanation completion and explanation-based learning

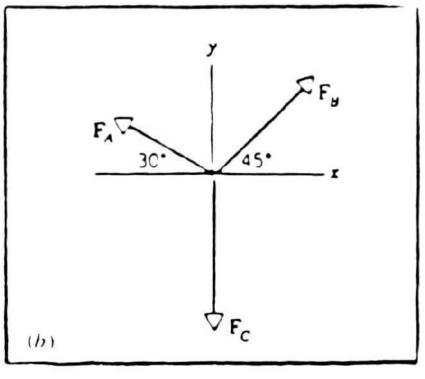
Because the Good students seem to learn primarily while explaining examples, two types of machine learning are likely candidate models: explanation-based learning (Mitchell, Keller & Kedar-Cabelli, 1986; DeJong & Mooney, 1986) and explanation completion (VanLehn, 1987; Schank, 1986). Explanation-based learning (EBL) is a large class of machine-learning algorithms that work as follows. The machine has a domain theory that is sufficient to solve a wide class of problems. However, the theory is expressed in a way that makes solution of most problems computationally infeasible. Although the theory can in principle solve any problem in its domain given infinite computational resources, it can in practice solve only a small subset of the problems. EBL increases the performance of the system as measured by the number of problems it can feasibly solve, but EBL does not change the set of problems that it can solve in principle. Explanation completion (EC), on the other hand, is a class of learning algorithms that increase the set of problems that are solvable in principle. To put it in once-fashionable terms, EBL increases the performance of the system but not its competence, while EC increases its competence. They can be used together to increase both competence and performance.

Both EBL and EC require as input an example consisting of a problem and its solution. Figure 1 shows one of the examples used in the Chi study. The statement of the problem consists of the first paragraph and the accompanying diagram. They state the physical situation and the sought quantities. The solution consists of the force diagram, the equations and the rest of the text. The solution is not a complete description of the reasoning processes required to solve the problem. For instance, there is no explanation of the minus sign in the equation $F_{Ax}=-F_A\cos(30°)$. In problem-solving terms, an example's solution presents only some of the intermediate states along the solution path, not all of them.

Figure 1: A example from Chi's self-explanation study

Figure 1a shows an object of weight W hung by massless strings. Consider the knot at the junction of the three strings to be "the body" The body remains at rest under the action of the three forces shown in Fig. 1b. Suppose we are given the magnitude of one of these forces. How can we find the magnitude of the other forces?





$\mathbf{F_A}$, $\mathbf{F_B}$, and $\mathbf{F_C}$ are *all* the forces acting *on* the body.

Since the body is unaccelerated, $\mathbf{F_A} + \mathbf{F_B} + \mathbf{F_C} = 0$

Choosing the *x*- and *y*-axes as shown, we can write this vector equation as three scalar equations:

$$\mathbf{F_{Ax}} + \mathbf{F_{Bx}} = 0,$$
$$\mathbf{F_{Ay}} + \mathbf{F_{By}} + \mathbf{F_{Cy}} = 0,$$

using Eq. 5-2. The third scalar equation for the *z*-axis is simply:

$$\mathbf{F_{Az}} = \mathbf{F_{Bz}} = \mathbf{F_{Cz}} = 0.$$

That is, the vectors all lie in the *x*-*y* plane so that they have no *z* components.

From the figure we see that

$$\mathbf{F_{Ax}} = -\mathbf{F_A}\cos 30^\circ = -0.866\mathbf{F_A},$$
$$\mathbf{F_{Ay}} = \mathbf{F_A}\sin 30^\circ = 0.500\mathbf{F_A},$$

and

$$\mathbf{F_{Bx}} = \mathbf{F_B}\cos 45^\circ = 0.707\mathbf{F_B},$$
$$\mathbf{F_{By}} = \mathbf{F_B}\sin 45^\circ = 0.707\mathbf{F_B}.$$

Also,

$$\mathbf{F_{Cy}} = -\mathbf{F_C} = -\mathbf{W}$$

because the string *C* merely serves to transmit the force on one end to the junction at its other end.

Substituting these results into our original equations, we obtain

$$-0.866\mathbf{F_A} + 0.707\mathbf{F_B} = 0$$
$$0.500\mathbf{F_A} + 0.707\mathbf{F_B} - \mathbf{W} = 0$$

If we are given the magnitude of any one of these three forces, we can solve these equations for the other two.

For example, if $\mathbf{W}$ = 100 N, we obtain $\mathbf{F_A}$ = 73.3 N and $\mathbf{F_B}$ = 89.6 N.

The first step in both EBL and EC is to explain the given example. Explaining an example is just like solving a problem, but is much easier computationally because the final state and some of the intermediate states are available.

EBL requires that an explanation be found for the example. The structure of the explanation depends on the kind of problem solving used for the task domain. It might be a solution path consisting of complete list of all intermediate states. It might be a solution tree, consisting of the goals, subgoals and primitive operators applied in generating the solution. It might be a causal network of instantiated schemata. The large variety of problem-solving techniques necessitates a large variety of explanation structures, and hence a large variety of EBL algorithms. Once an explanation structure has been found, the EBL algorithm analyzes it in order to find methods to speed up future problem solving on this kind of problem. There are a variety of methods for speeding up problem solving, so there are also a variety of analytic techniques used by EBL algorithms.

EC requires that only a *partial* explanation be found for the given example. For instance, the solver might be able to explain everything in Figure 1 except for the minus sign of $F_{Ax}=-F_A\cos(30°)$. The job of EC

is to invent a new piece of knowledge that will complete the partial explanation. The new piece of knowledge should be plausible, and the various ways of defining "plausible" lead to various kinds of EC algorithms. Sierra (VanLehn, 1987) evaluated the plausibility of proposed completions syntactically, by measuring their size. This implemented a form of Occam's Razor, which suggests that the simplest hypothesis is the best, all other things being equal. Other EC programs have also used syntactic plausibility criteria successfully (e.g., Berwick, 1985; Hall, 1988). Schank (1986) pioneered the use of knowledge-based evaluations of plausibility. Completions are more plausible if they are consistent with known explanation patterns. Schank's task domain is common sense theories of the world, which explain, for instance, why a famous race horse might die at the height of his career. For instance, one of Schank's explanation patterns is "killed for the insurance money." Lewis (1988) and Anderson (1989) use explanation patterns taken from the causal attribution literature, because their domain theories concern the operation of physical devices. The variation among EC algorithms stems mainly from whether they use syntactic or knowledge-based definitions of plausibility, and from the domain-dependent nature of plausibility.

Both EBL and EC are consistent with the main findings of the self-explanation studies. Both learning techniques require explanation of the examples, so the Poor students, who did not explain the examples, could not engage in these learning processes and hence should learn less. This explains the first finding, which is that the amount of explanation given examples correlates with the amount of learning. The second finding is that the Good students tend to say, "I don't understand that," whereas the Poor Students tend to say, "Yes, that makes sense." If EBL improves problem solving, then the problem solving must be non-optimal as the student tries to explain the example. Thus, there must be some investigation of false paths during example explaining. Assuming that the learning process is EBL explains why the Good students often make negative self-monitoring statements. EC can also explain this finding, for it requires that explanations be only partial. Failing to explain a part of the example is what causes the Good student to say, "I don't understand that." In summary, it seems that both correlations seen in the self-explanation studies are adequately explained by both EC and EBL. A finer analysis of the data is necessary in order to tell whether the source of the self-explanation effect is EBL, EC, both or neither.

The analysis presented here shows that neither EBL nor EC is adequate for explaining the self-explanation effect. Instead, we propose a new learning technique, *explanation-based learning of correctness* (EBLC), that seems to be a better model of the self-explanation effect than EBL and/or EC. The following sections discuss the inadequacies of EBL and EC, then present EBLC and show how it overcomes these inadequacies.

Explanation-based learning as an account of self-explanation

If EBL is to account for the self-explanation effect, then students would have to acquire a complete domain theory before studying the examples. There are two reasons to believe that this could not have occurred. First, Chi et al. (1989) tested students' knowledge of Newton's laws just before they started studying the examples. Good and Poor students knew only 5.5 of the 12 components that constitute a complete understanding of the laws, according to expert physicists. Moreover, Good students gained 3.0 components during the example studying, whereas the Poor students gained only 0.25 components. These findings are not consistent with the assumption that the students had a complete understanding of the domain before studying the examples.

The second reason for doubting that students acquired a complete domain theory prior to studying the examples is that the examples present important information that is just not presented anywhere earlier in the text. For instance, the concept of normal force is presented for the first time in the examples. Later we shall present an analysis showing that only 11 of the 28 physics rules needed for solving the Chi problems are presented in the chapter. The other 17 are presented for the first time in the examples. Basically, the chapter devotes most of its prose to the experimental evidence for Newton's laws and their history. Problem solving is taught by example.

In short, the hypothesis that EBL alone can explain the self-explanation effect is inconsistent with the details of the experimental and textual evidence. Thus, the cause of the effect must be either EC, a combination of EC and EBL, or something new.

Explanation completion in Cascade

The next step in the argument is to assess whether EC can explain the self-explanation effect. To do so, we constructed Cascade1, a simple problem solver that learns via explanation completion. As will be seen later, the particular form of EC used in Cascade1 forms the basis of explanation-based learning of correctness (EBLC), which is the learning technique that seems most likely to be the cause of the self-explanation effect. Thus, we will discuss Cascade1 in some detail.

Because Cascade1 only tests EC and not EBL, it does not need the models of memory, elementary processes, and so forth that a cognitive performance model requires. Therefore we chose a problem-solving framework that is easy to work with even though it is not plausible as a cognitive architecture. Cascade1 represents knowledge as Horn clauses and uses backwards chaining as it sole problem-solving method. Thus, Cascade1 started out as a version of pure Prolog. (Descriptions of this type of problem solver can be found in most recent AI textbooks, e.g., Charniak & McDermott, 1986.) As discussed later, extra mechanisms were added in order to make EC work correctly.

It turned out to be quite simple to use this problem solving architecture to explain examples, rather than solve problems. During problem solving, the problem's givens are encoded as facts and the sought values are specified as a goal. By deriving (proving) the goal from the facts, Cascade produces values for the sought quantities. For instance, for the problem in Figure 1, the top level goal would be:

F=magnitude(force(block,string,tension)), and
N=magnitude(force(block,plane,normal))

In the process of proving these statements, specific values for the variable F and N would be calculated. The simplest way to model the explanation of an example is to ask Cascade1 to prove that the final answer is correct. Thus, the following would be used for the example of Figure 1:

mass(block)*g*sin(angle(plane,horizontal))=magnitude(force(block,string,tension)), and
mass(block)*g*cos(angle(plane,horizontal))=magnitude(force(block,plane,normal))

Specific expressions have been substituted for F and N in the top level goal. But this formalization of example explaining is too simple, for it omits all the intermediate state information given by the example. Therefore we represent the example as a sequence of subproblems, such as generating the free-body diagram, generating the vector equation, choosing coordinate axes and generating the scalar equations. Explaining an example is modeled by having Cascade1 prove correct the example's answers to each of these subproblems.

We invented a very simple type of EC for Cascade1. Whenever the backwards chainer fails to prove the current goal (which could be a subgoal of some higher goal) using the regular rules, it tries to prove it using special rules that correspond to Schank's explanation patterns. We found that only two of these special rules were necessary. The first rule, *Generalized Distribution*, is an overgeneralization of the arithmetic principle of distribution. For instance, it would sanction both $a(b+c)=ab+ac$ and $log(b+c)=log(b)+log(c)$ even though the latter is incorrect. The second rule, *Property Value Conservation*, comes from a general property of mathematical calculations, which is that they tend to move symbols around but not destroy them. The particular case addressed by the rule involves symbols that are the values of an object's properties. The rule states that if initially P(x)=V, where x is some object, P is a property and V is a value, and later it is observed that Q(x)=V, then it is likely that P(x)=Q(x). That is, the explanation for Q(x) having value V in the example is that Q(x) always has the same value as P(x) and that P(x)=V in the example.

Adding this mechanism of EC to the Horn clause architecture was very simple. It required only two changes. In order to make the rules general, we had to allow pattern-matching variables in predicate positions and function positions. That is, in order to get the overgeneralized distribution rule to match both $\log(ab)$ and $\sin(ab)$, we had to use a pattern-matching variable in a function position, where pure Prolog would only allow a pattern-matching constant. Secondly, in order to have the application of these rules actually change the rules that represent the subjects' physics knowledge, we had to have them write a new rule into the rule base. This was accomplished by adding a version of Prolog's "assert" predicate. Testing this predicate causes its argument, a new rule, to be added to the rule base.

One mechanism that we did not need was any kind of meta-level demon that would watch for failures, wrest control from the normal interpreter just before it was about to backup, and cause the special rules to be run instead. Instead, we just added the special rules at the end of the list of rules. Because Cascade1 tries the rules in order, the special rules would be tried only after all the normal rules had been tried. If they too failed, only then would Cascade1 back up.

It rapidly became clear that search had to be very carefully controlled if this simple approach to EC was to work. Every time the normal rules fail, Cascade1 tries to learn. Sometimes it would learn new clauses even when we didn't want it to. The Property Value Conservation rule was particularly nasty at this. If Cascade1 went down a bad search path, there was a good chance that the rule would postulate a spurious connection between two property values, causing the search to succeed when it shouldn't have.

Although we made several changes in order to control the search, the most effective and interesting one was to add caching to Cascade1. Whenever Cascade1 proved a goal, it would be added to the front of

the list of facts (normally, the list of facts contains only the information given in the problem). Caching does not change the competence of Cascade1, but it does change the order in which it investigates search paths. When trying to prove a goal, it first checks the fact list, starting at the front. Putting new statements at the front of the fact list gives Cascade1 a recency effect. Statements that it has been thinking about recently will be considered first when trying to prove a new goal. In this task domain, recency seems to help Cascade1 take the correct search path first. This in turn reduced the amount of mislearning.

Computational sufficiency

Cascade1's EC technique is so simple that there is some doubt as to whether it is computationally sufficient to learn all the rules that a subject might learn while explaining examples. In order to test its computational sufficiency, we put Cascade1 to the following test.

First, we developed a set of rules sufficient to correctly solve all the problems in the Chi experiment. We tried to make this rule set a faithful representation of the Good students' knowledge by comparing its behavior to the behavior of the Good students on the final problems in the Chi experiment. There were 111 rules in this set.

Next, we classified each rule according to where it could have first been learned, using the following classifications. The numbers in parentheses are the number of rules in each classification.

- Text (11). Mentioned in the textbook prior to the examples.
- Examples (17). Used for the first time in the examples.
- Math (50). Learned before the physics study.
- Common sense (17). Learned before the physics study.

In addition, 16 rules were classified as non-knowledge, since they merely changed the format of various data structures without modifying their content. They are artifacts of the formalization.

Next, we determined whether Cascade1 could learn each of the 17 rules that are used for the first time in the examples. These rules are listed in table 1. The testing procedure was simply to delete the rule from the rule set, have Cascade1 explain the examples, and see if it regenerated the rule.

Table 1: Rules to be learned while explaining examples

1. If all the forces on a body are collinear, then the acceleration of the body must be along that line.
2. If a body is sliding along an inclined plane, then the acceleration of the body is parallel to the plane.
3. If a body is supported only by an inclined plane, the acceleration has a downwards sense.
4. A normal force is perpendicular to the surface causing it.
5. A normal force points up.
6. If a string has parts, then a tension force on a body caused by the string is parallel to the part of the string attached to the body.
7. If a string has no parts, then a tension force on a body attached to the string is parallel to the string.
8. If a string pulls up on a body, then the tension force on the body has an upwards sense.
9. If a string pulls down on a body, then the tension force on the body has a downwards sense.
10. The magnitude of a tension force caused by a string is equal to the string's tension.
11. If two blocks hang from a string that runs over a pulley, then their accelerations are equal in magnitude.
12. If two blocks hang from a string that runs over a pulley, then their accelerations have opposite senses: one up and the other down.
13. A knot can be a body.
14. Projection(L=R,axis) is Projection(L,axis)=Projection(R,axis). Projection onto an axis distributes over equations.
15. If S is a scalar and V is a vector, then Projection(S*V,axis) is S*Projection(V,axis).
16. When all forces on a body are collinear, the free-body diagram has only one axis.
17. When two forces on a body are perpendicular, the free-body diagram should have an axis aligned with each of them.

Generalized distribution was able to learn rule 14, and a small modification of it would suffice to learn rule 15. These are the only two rules that have the right form for learning by Generalized distribution. Many rules have roughly the right format for them to be learned by Property Value Conservation, but Property

Value Conservation was able to learn only rules 6, 7 and 10. It's inability to learn rules 4, 8 and 9 are due to its inability to calculate one value given another. For instance, rule 4 needs to assign a value to the inclination of a normal force, but the value it assigns is not simply the value of an existing property. Rather, the normal force's inclination is 90 degrees plus the plane's inclination. Although Property Value Conservation is not powerful enough to learn these rules, it is clear how to create a new explanation pattern that could. The basic mechanism of EC is not threatened by the inability of Cascade1 to learn these three rules.

Rules 1, 2, 11, 12 and 17 present a more difficult problem. Their conditions are too complicated for Property Value Conservation to regenerate. Property Value Conservation would produce a subset of the rule's conditions, thus creating an overly general rule. Rules 3, 5, 13 and 16 have this same problem and another one as well. They assign a value to a property that does not come from some pre-existing property's value. They violate the essential feature of Property Value Conservation, which is that value symbols are conserved, by creating a value out of thin air, as it were. Altogether, these 9 rules indicate that there are serious problems with Cascade1's learning.

Upon reflection, it seems that the culprit is the explanation-pattern approach to EC. An explanation pattern, especially as implemented in Cascade1, is just another rule, albeit an overly general, possibly incorrect one. If explanation patterns are viewed as part of the domain theory, then this approach to learning is really a form of EBL. Seen this way, the underlying problem in Cascade1 becomes apparent. It expects to learn from the application of a single rule. That is like insisting that EBL only learn from a one-step explanation. Clearly, this is not powerful enough. For instance, one way to learn rule 2 is to reason as follows:

> (1) If a body slides along a surface, then its velocity is parallel to the surface. (2) If a body slides along a flat surface, then its velocity is straight. (3) An inclined plane is a flat surface. (4) If the velocity of an object is straight, then its acceleration is parallel to its velocity. (5) If A is parallel to B and B is parallel to C, then A is parallel to C. From these 5 statements, it follows that: if a body slides along an inclined plane, then its acceleration is parallel to the plane.

This multi-step explanation requires quite a bit of common sense and mathematical knowledge, so it is unlikely that a single explanation pattern would exist for it. However, because common sense, and naive physics in particular, can be notoriously inaccurate, such multi-step explanation retain an important property of explanation patterns, which is that they are not guaranteed to be correct. Indeed, our intuition is that most students who explain rule 3 would leave out the stipulations involving flatness and straightness. This would produce the same correct conclusion even though it uses incorrect rules.

Thus, the right way to view this form of learning is that the machine first tries to use correct knowledge, but when that fails to explain something, it uses a mixture of correct and highly general, possibly incorrect knowledge. If this succeeds, it stores the explanation away for use in later problem solving. The usual EBL processes can be used to replace some of the constants appearing in the explanation with variables, thus generalizing the explanation into a rule. Presumably, if this new rule is often used with success in later problem solving, the machine's estimate of its correctness will gradually increase.

This learning process is not exactly EBL. In EBL, the net effect is an increase in the speed or efficiency of the computation. Although this may also occur in this learning process, the more important effect is an increase in the number of problems that can be answered with high probability of a correct answer. As far as we know, this is an entirely new type of machine learning, so we have given it a name: *explanation-based learning of correctness* or EBLC. It clearly deserves more exploration. We plan to do so in the context of the next version of Cascade.

In summary, we have found that Cascade1's approach to EC, which was based on one-step application of explanation patterns, was not computationally sufficient to learn all the rules that subjects might learn from the examples. However, we discovered a new learning process, EBLC, which is both a kind of EC and a kind of EBL. This new learning technique seems able to learn all the rules, and plans are being made to test it.

Protocol evidence for the explanation completion

This section analyzes protocol data as a second argument to show that explanation completion is not a good account for the self-explanation effect. Along the way, more evidence is accumulated about the nature of self-explanation that will shape the final definition of EBLC, which is presented in the section following this one.

In both Cascade1's EC as well as any other kind of EC, the missing rules in the knowledge base determine which parts of an example cannot be explained. Thus, we can test the hypothesis that EC

underlies the self-explanation effect without knowing the precise details of the EC process. We need only examine the parts of the protocol where an EC process would have to occur, given our assumptions about which rules are missing from the subject's knowledge. This section discusses such an analysis.

We analyzed the protocol of one subject, S101, who was a particularly articulate Good student, as he studied examples and solved problems. For each of the 17 rules in table 1, we expected to find some sign of rule acquisition in at least one of the places where that rule could potentially be used. Protocol analyses by ourselves and others (VanLehn, 1989a; VanLehn, 1989b; Siegler & Jenkins, 1989) found that 90% of the places where a rule was acquired (as determined by a shift in the subject's later behavior), the protocol was marked either by a long pause, on the order of 20 seconds or more, or by extended comments about the discovering activity. Using these criteria, we located each place in the protocol where one of the 17 rules could be used and carefully analyzed the text in that vicinity. We also examined the rest of the protocol looking for obvious signs of rules being learned.

We found 5 clear cases of rules being discovered in the S101 protocol. However, only one of these 5 rule acquisition events corresponded to any of the 17 rule acquisition events predicted by the EC hypothesis. Of the remaining 4 cases, one resulted in the acquisition of an incorrect, buggy rule and 3 involved acquisition of qualitative physics rules. Because the Cascade1 simulation modeled only the acquisition of quantitative, correct knowledge, it did not predict 4 of the 5 observed learning events. By tuning the Cascade1 simulation to S101's prior knowledge and simulating his thirst for a qualitative explanation as well as a quantitative one, it may be possible for Cascade1 to account for all 5 of S101's learning events, so we do not view this aspect of the mismatch between data and predictions as particularly informative. In particular, it doesn't tell us much about the ability of EC to model S101's learning.

The more informative result is that only one of the predicted rule acquisition events showed up in the protocol data. At all the other places, the subject just used the rule with no more commenting nor pausing than he typically exhibited during other parts of the protocol. Although the informality of this analysis makes the finding quite tentative, it is consistent with the hypothesis that the first use of 16 of the 17 rules evoked no more cognitive processing than ordinary rule application would.

It could be that there was no sign of learning these 16 rules because the rules were already known. Perhaps those 16 rules are actually presented by the text before the examples, and we overlooked them. We reanalyzed the chapter looking specifically for those rules, and could find no signs of them. Another possibility is that the subject learned these 16 rules in his high school physics course. However, it would be unlikely for him to recall so many of them, especially when the scores from the test on the components of Newton's laws indicated grave deficiencies in Good student knowledge prior to studying examples. If the subject did not learn the rules before the experiment and did not learn them while reading the text that precedes the examples, then he must have learned the rules during the examples using a learning process that does not generate long pauses or unusual comments in the protocols.

Conclusions

The findings from both the computational sufficiency analysis and the protocol analysis can be explained if self-explanation is modeled by explanation-based learning of correctness (EBLC). EBLC is based on the following three assumptions. (1) Rules are not dichotomized as correct, proper components of the domain theory versus incorrect, heuristic, overly general explanation patterns. Instead, the correctness is a continuous quantity. (2) Learners prefers to use more correct rules before less correct ones. If they are forced to use rules with very low correctness values, then they might complain, pause or show other signs that would normally be interpreted as a problem solving impasse. (3) Whenever a rule is used in a successful explanation, a more specific version of the rule is created and stored in memory. The correctness value of the new rule is higher than the old one because it participated in a successful explanation.

EBLC blurs the distinction between EBL and EC by blurring the distinction between a complete explanation and a partial one. According to EBLC, students estimate an explanation as more or less correct, depending on which rules are used in forming it. When students get feedback from the textbook or a teacher on the actual correctness of an explanation, they update their estimates of the correctness of the rules used. The simplicity and rationality of this model is rather compelling.

Moreover, it makes sense of the protocol findings. S101 did not pause or complain when applying 16 of the 17 rules that should have been first generated during the explanation of examples. We can account for this by assuming that he didn't apply the rules per se, but used general, common sense reasoning instead. However, these common sense rules were apparently correct enough that the subject did not pause or complain.

Stepping back to look at the overall argument, we have achieved our original goal of determining whether self-explanation was primarily due to EBL, EC or something else. The most plausible hypothesis is

that self-explanation is due to EBLC, which is both a type of EBL and EC. Moreover, from a computer-science perspective, we have come to the novel (to us at least) conclusion that when a learner uses *plausible reasoning* rather than logically correct reasoning, then EBL and EC are actually the same processes. This conclusion may shed new light on other parts of cognitive science. For instance, it blurs the distinction between competence or knowledge-level learning (such as EC) and performance or symbol-level learning (such as EBL).

Acknowledgments

This research was supported by the Office of Naval Research's Cognitive Sciences Program, contract N00014-88-K-0086 and by the Office of Naval Research's Computer Sciences Division, contract N00014-86-K-0678.

References

Anderson, J.R. (1989). A theory of the origins of human knowledge. *Artificial Intelligence*, *40*(1-3), 331-352.

Berwick, R. (1985). *The Acquisition of Syntactic Knowledge*. Cambridge, MA: MIT Press.

Charniak, E. & McDermott, D. (1986). *Introduction to Artificial Intelligence*. Reading, MA: Addison-Wesley.

Chi, M.T.H., Bassok, M., Lewis, M., Reimann, P. & Glaser, R. (1989). Self explanations: How students study and use examples in learning to solve problems. *Cognitive Science*, *13*, 145-182.

DeJong, G. & Mooney, R. (1986). Explanation-based learning: An alternative view. *Machine Learning*, *1*(2), 145-176.

Ferguson-Hessler, M.G.M. & de Jong, T. (1990). Studying physics tests: Differences in study processes between good and poor performers. *Cognition and Instruction*, *7*(1), 41-54.

Hall, R. (1988). Learning by failing to explain: Using partial explanations to learn in incomplete and intractable domains. *Machine Learning*, *3*(1), 45-78.

Lewis, C. (1988). Why and how to learn why: Analysis-based generalization of procedures. *Cognitive Science*, *12*, 211-256.

Mitchell, T.M., Keller, R.M. & Kedar-Cabelli, S.T. (1986). Explanation-based generalization: A unifying view. *Machine Learning*, *1*(1), 47-80.

Pirolli, P. & Bielaczyc, K. (1989). Empirical analyses of self-explanation and transfer in learning to program. In G. Ohlson & E. Smith (Ed.), *Proceedings of the Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Erlbaum.

Schank, R.C. (1986). *Explanation Patterns: Understanding mechanically and creatively*. Hillsdale, NJ: Erlbaum.

Siegler, R.S. & Jenkins, E.A. (1989). *How children discover new strategies*. Hillsdale, NJ: Erlbaum.

VanLehn, K. (1987). Learning one subprocedure per lesson. *Artificial Intelligence*, *31*(1), 1-40.

VanLehn, K. (1989). Rule acquisition events in the discovery of problem solving strategies. Submitted for publication. Currently available as technical report PCG-17, Dept. of Psychology, Carnegie-Mellon University.

VanLehn, K. (1989). Learning events in the acquisition of three skills. In G. Ohlson & E. Smith (Ed.), *Proceedings of the Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Erlbaum.

Indexing Libraries of Programming Plans

James C. Spohrer

Apple Computer, Cupertino, CA

ABSTRACT

As student programmers learn to write programs, they acquire programming plans. The plans must be indexed or organized in some manner that allows the student to access "the right plan at the right time." In this paper a preliminary indexing scheme is proposed that is based on observations from thinking-aloud protocol data collected as students wrote introductory Pascal programs.

INTRODUCTION: MOTIVATIONS AND GOALS

Programming plans are commonly used lines of computer instructions that work together to achieve programming goals (e.g., getting valid input data, reading in a series of values, calculating results based on input values, etc.). Student programmers acquire plans while solving programming tasks, gradually building up an internalized library of plans that they then reuse to solve new tasks. The cognitive plausibility of specific programming plans has been established in a number of studies [Sh76] [MRRH81] [Ad81] [BS83] [SE84] [R86].

Recently, student programming environments have been constructed that allow students to explicitly construct plans, save them in an on-line library, and then reuse them to solve new tasks (Soloway, personal communications). However, two major problems arise: (1) it is difficult to index or organize the plans in such a way that students can easily find and reuse plans, and (2) many bugs arise when students try to compose plans into larger structures.

In this paper, one possible indexing scheme is proposed for a class of programming plans. The indexing scheme presented in this paper is part of a larger theory of the way students write and debug programs [S89]. The indexing scheme has been used in the MARCEL program [S89], which is implemented on Apple Macintosh II computers and simulates students writing Pascal programs. In the next section, the type of programming tasks (arithmetic word programming tasks) for which the proprosed indexing scheme was developed will be described. Next, the indexing scheme will be presented. In conclusion, the implications of this indexing scheme for programming instruction will be discussed.

The Subjects and the Tasks

Yale University undergraduates who were taking their first introductory Pascal programming class were asked to solve three programming tasks:

1. The Electric Bill Task: Calculate the electric bill for a customer (*id*) based on how many kilowatt hours of electricity the customer used (*kwh*). The charge is 9 cents for the first 350 kwh used, 5 cents for the next 275 kwh used, 4 cents for the next 225 kwh used, and 3 cents for all usage over 850 kwh. [Implicit requirement from classroom lecture: The program should print an error message and stop if the kwh amount input is invalid.]

```
input(id,kwh)
if invalid
   then error
   else begin
          calculation
          output(id,kwh,cost)
        end
```

2. The Reformatting Task: Read in *raw data* collected during an experiment and print out the *reformatted data*. The input is the subject number, problem type ('a', 'b', or 'c'), the start and end times of the experiment (in hours, minutes and seconds), and the subject's accuracy ('+' or '-'). The output should be the subject number, the problem type, the elapsed time in seconds, and the accuracy. Perform *valid-data-entry*: If any of the input values are invalid, give the user a second chance to enter them and assume valid data will be entered the second time. The program should input and print out data for a series of subjects as long as the user has *more data* to process; stop when a *sentinel* value is entered.

```
valid-data-entry(more-data)
while not-sentinel(more-data)
  do begin
        valid-data-entry(raw-data)
        calculation
        output(reformatted-data)
        valid-data-entry(more-data)
     end
```

3. The Rainfall Task: Read in a series of rainfall values stopping when the *sentinel* value (999999) is entered. The sentinel value is a special value, indicating the end of the input data, and should not be included in the calculation. If the input is invalid, prompt the user again and again until the input is valid (*vde*). The program should print out the number of valid rainfall amounts entered, the number of rainy days, the maximum rainfall amount, and the average rainfall amount.

```
initialization;   vde(rain)
if sentinel(rain)
   then no-valid-input-error
   else begin
          while not-sentinel(rain)
            do begin
                 update; vde(rain)
               end
            calculation; output
        end
```

Figure 1: Three tasks and pseudo-code programs.

The three tasks are representative introductory programming tasks requiring conditionals and loops, but not requiring arrays or procedures. *Arithmetic word programming tasks* resemble non-programming arithmetic word tasks [KG85] [MPS87].

Two types of plan knowledge are equired to solve arithmetic word programming tasks like those studied: (1) communication plans, and (2) calculation plans (see Figure 2).

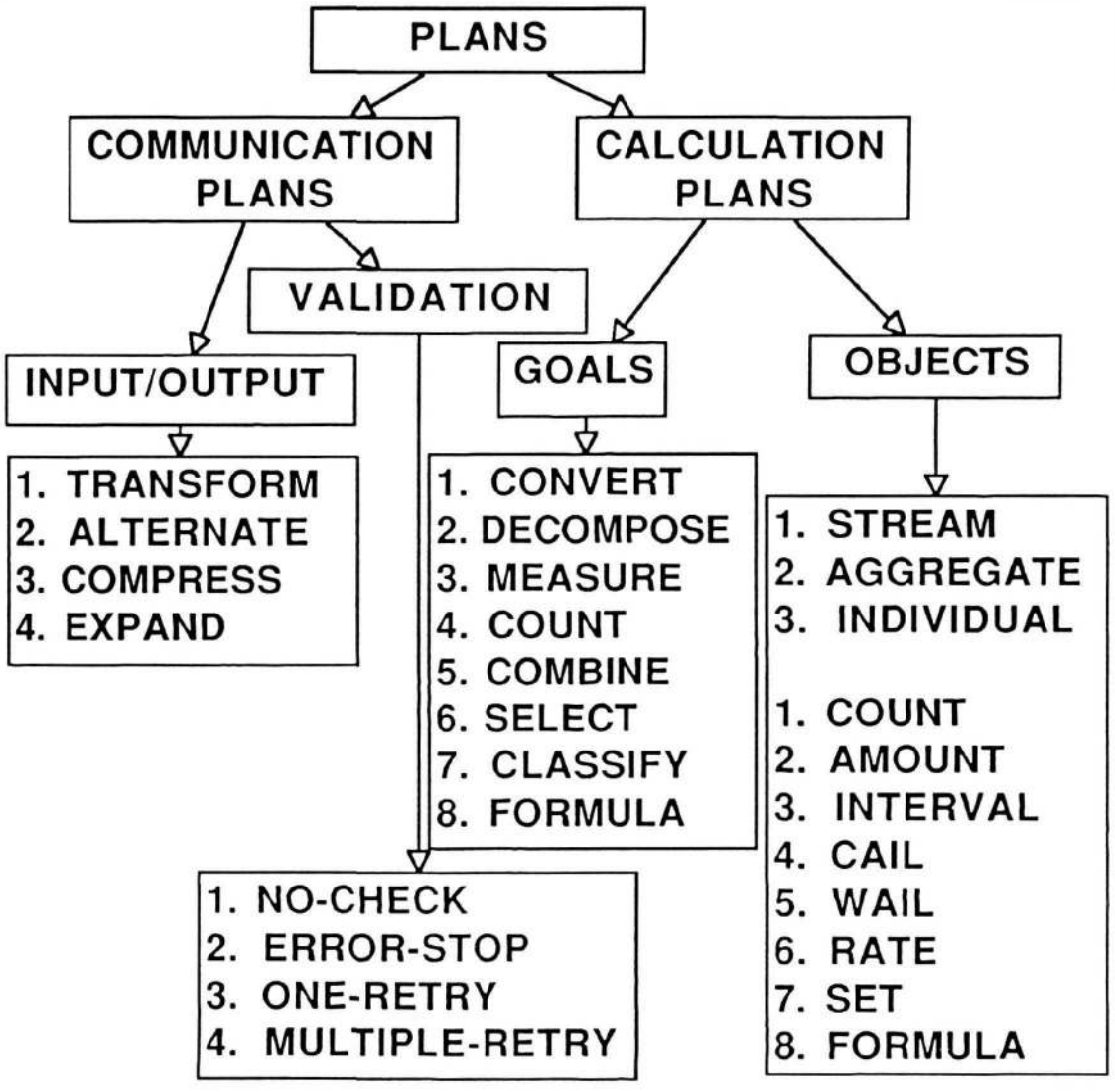


Figure 2: Programming plan indices for arithmetic word tasks.

Students possess well developed prior knowledge or *preprogramming knowledge* in each of these two areas: (1) students reason about communication between agents when they write programs [P86], and (2) students, using paper and pencil, can easily solve the underlying calculation part of the tasks.

Communication Plans

As shown in Figure 2, the communication plans category splits into two subcategories: (1) input/output plans, and (2) input validation plans.

Communication Plans: Input/Output Plans

The four major types of input/output plans are:

1. Transform: Input is non-stream, and output is non-stream.
2. Alternate: Input is stream, and output is stream.
3. Compress: Input is stream, and output is non-stream.
4. Expand: Input is non-stream, and output is stream.

A *stream* is a series of values (e.g., find the sum of a series of numbers). A *non-stream* is either an *individual* (a single value) or an *aggregate* (a small fixed set of values).

Communication Plans: Validation Plans

The four major types of validation plans are:

1. No-Check: Do nothing about invalid input; the input is not checked.
2. Error-Stop: If the input is invalid, then print an error message, and stop.
3. One-Retry: Print an error message, and give the user one more chance to enter a valid value. Assume the retry is valid.
4. Multiple-Retry: Print an error message, and give the user as many chances as needed until the input is valid.

The plans vary in whether validation is done at all, and if so, how many retry opportunities (zero, one, or multiple) are supported.

Communication Plans: Combined I/O andValidation Plans

Figure 1 shows pseudo-code programs for three of the possible combined I/O and validation plans. In the Electric Bill Task, the input is the amount of electricity used (non-stream), the output is an electric bill (non-stream), and the validation requires stopping on error, so that the communication plan for this task is a Transform/Error-Stop plan. In the Reformatting Task, the input is a series of start and end times (stream), the output is a series of elapsed times (stream), and the validation rquires one retry on errors, so that the communication plan for this task is an Alternate/One-Retry. In the Rainfall Task, the input is a series of rainfall amounts (stream), the output is an average (non-stream), and the validation is multiple retries on errors, so that the communication plan for this task is Compress/Multiple-Retry.

Calculation Plans

The calculation plans are indexed by a small set of goals and objects:

G1. Convert: The convert goal is used to change an amount in one unit (e.g., Fahrenheit) to an different unit (e.g., Celsius).

G2. Decompose: The decompose goal is used to break an amount into parts based on some criteria (e.g., 325 = 3*100 + 2*10 + 5).

G3. Measure: The measure goal is used to find the third part of an interval (e.g., size) from the other two parts (e.g., start,end).

G4. Count: The count goal is used to count the items in a stream.

G5. Combine: The combine goal is used put parts together into a whole amount, or stream items together into a new amount.

G6. Select: The select goal is used to select an item from a stream based on some criteria (e.g., maximum).

G7. Classify: The classify goal is used to filter a stream into substreams, or determine if an object is in a "special" state.

G8. Formula: The formula goal is a computation involving an arbitrary use of arithmetic operations (e.g, average).

O1. Count: Count objects correspond to the number of items in a stream, or the position of an item in a stream.

O2. Amount: Amount objects are non-negative quantities measured in some units (e.g., amount of rainfall in inches).

O3. Interval: Interval objects are used to describe the relationship between three amount objects that correspond to start, end, and size (e.g., starttime, endtime, and elapsed time).

O4. CAIL: CAIL objects with ceiling and overflow computation (see [S89]).

O5. WAIL: WAIL objects deal with wrap-around or MOD arithmetic (see [S89]).

O6. Rate: Rate objects are amounts that can be used to convert other amounts into different units (e.g., hours into minutes).

O7. Set Set objects can take on one of a fixed (usually small) number of values (e.g., '+' or '-' for correctness or accuracy).

O8. Formula: Formula objects result from calculations involving formulae (e.g., in average = sum/count, average is a formula object, sum is an amount object, and count is a count object).

The goals and objects were extracted from the protocol data (see next page).

Support for Plausibility Claim

Protocol snippets for all of the goals and objects are shown below.

G1. CONVERT: *"Now I'm ready to start converting in... the seconds" (AVM6.239).*

G2. DECOMPOSE: *"I'm gonna try to break up the kilowatt hours into under 350... part of them are going to be less than 350." (JBH5.58-68).*

G3. MEASURE: *"Convert the starting time and finishing time to numbers in seconds, then subtract to find the number of seconds difference..." (AAS6.118).*

G4. COUNT: *"Right there, I want to keep track of how many days that go by where there is no rain, when it's equal to zero.* [Experimenter: Right. And what's the problem?] *That I don't know how to count." (JBH7.72-74).*

G5. COMBINE: *"I still want to keep track of how much rain fell in all." (JBH7.122).*

G6. SELECT: *"If it's bigger than the maximum value already in, then the maximum value is gonna become that value as entered." (AAS7.28).*

G7. CLASSIFY: *"For each time the rainfall is less than zero..." (JBH7.60).*

G8. FORMULA: *"Ok I need to define uhm, avg is average rainfall per day. So its total amount of rain which is sum divided by how many days there were." (AAS7.33).*

O1. COUNT: *"I'm gonna... need to figure out how many values he's entered." (AAS7.6).*

O2. AMOUNT: *"It's gonna need one variable for the amount... it's gonna need the first variable for the amount of rain he types in." (AAS7.2).*

O3. INTERVAL: *"Convert the starting time and finishing time to numbers in seconds, then subtract to find the number of seconds difference..." (AAS6.118).*

O4. CAIL: *"I'm gonna try to break up the kilowatt hours into under 350... part of them are going to be less than 350." (JBH5.58-68).*

O5. WAIL: *"So, the first case is if the starting time isn't gonna roll over, cause then I can subtract the beginning from the end. If there's rollover... add... constant" (AAS6.122).*

O6. RATE: *"It seems like this will be some sort of rate problem." (JBH5.1).*

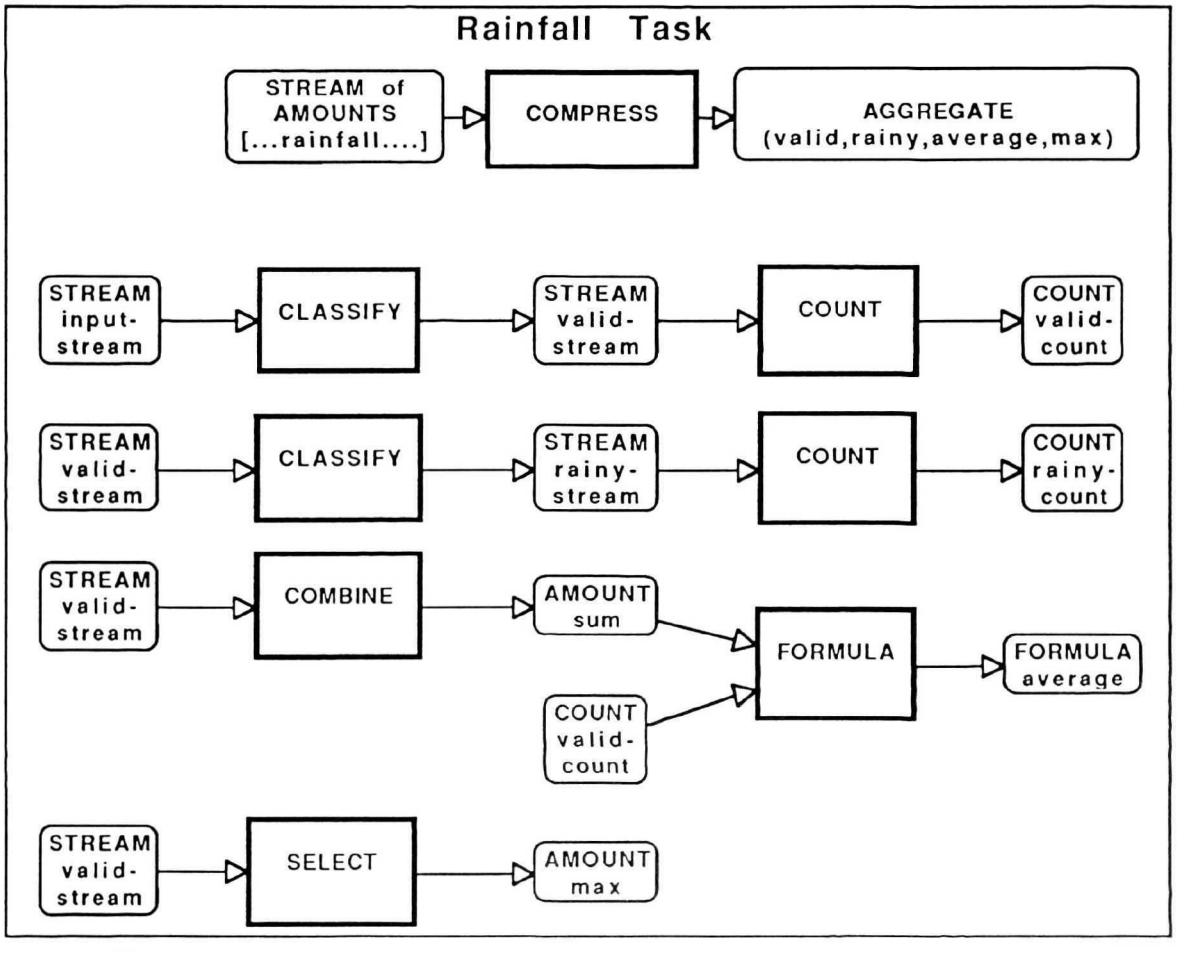
O7. SET: *"...the correct problem type, if it's not 'a', 'b' or 'c'." (AVM6.105)."*

O8. FORMULA: *"Ok I need to define uhm, avg is average rainfall per day. So its total amount of rain which is sum divided by how many days there were." (AAS7.33).*

Figure 4: Protocol evidence for conceptual goals and objects.

Representing Programming Tasks

An arithmetic word programming task can be presented in terms of the goals and objects that have been identified in the data. For instance, in the Rainfall Task, a stream of input "rainfall" values is "compressed" into an output aggregate consisting of "valid_count", "rainy_count", "average", and "maximum" -- summary statistics for the input stream. Each of the output objects requires a different goal-chain to produce it. Ultimately, the goal-chain traces back to the input stream of "rainfall" values. The "valid_count" object is a count object that is produced by a count goal consuming a "valid_stream" (see second graph from top in box). The "valid_stream" is a stream object that is produced by a classify goal consuming the original "input_stream". Similarly, the calculation of the other three objects in the output aggregate result from goal-chains composed of different intermediate goals and objects.



Concluding Remarks

In this paper a preliminary indexing scheme for libraries of programming plans has been presented. The indexing scheme has been successfully employed in MARCEL, a program that simulates student programmers writing and debugging Pascal programs. There are two main instructional implications of this work: (1) the indexing scheme and the plans indexed by the scheme might be explicitly taught to students with the hope of accelerating their learning rates while avoiding certain plan composition bugs, and (2) new programming environments might employ the indexing scheme to help students better organize their explicit plan knowledge and therefore be able to "access the right plan at the right time" more easily and accurately. A future research direction is to develop indexing schemes for additional types of programming tasks besides simple arithmetic word tasks.

References

[Ad81] Beth Adelson. Problem solving and the development of abstract categories in programming languages. *Memory and Cognition*, 9,422-433, 1981.

[BS85] J. Bonar and E. Soloway. Preprogramming knowledge: A major source of misconceptions in novice programmers. *Human-Computer Interactions*, 1(2):133-161, 1985. (Reprinted in [SS89]).

[KG85] W. Kintsch and J. G. Greeno. Understanding and solving arithmetic word problems. *Psychology Review*. 92, 109-129, 1985.

[MPS87] S.P. Marshall, C.A. Pribe, and J.D. Smith. *Schema knowledge structures for representing arithmetic story problems*. TR, Dept. of Psych., SDSU, 87.

[MRRH81] K.B. McKeithen, J.S. Reitman, H.H. Rueter, and S.C. Hirtle Knowledge organization and skill differences. in computer programmers. *Cog.Psych*. 13, 307-325, 1981.

[P86] R. Pea. Language independent conceptual "bugs" in novice programs. *Journal of Educational Computing Research*. 2(1):25-36,1986.

[R86] Robert S. Rist. Plans in programming: definition, demonstration, and development, pages 28-47. In *Empirical Studies of Programmers*, Soloway and Iyengar, Ed. Ablex, Norwood NJ, 1986.

[Sh76] B. Shneiderman. Exploratory experiments in programmer behavior. *Int. J. Comput. Inf. Sci.*. 5, 2, 123-143, 1976.

[SE84] E. Soloway, and K. Ehrlich. Empirical studies of programming knowledge. *IEEE Transactions on Software Engineering*. 5:595-609, 1984.

[SS89] E. Soloway and J.C. Spohrer, Editors. *Studying the Novice Programmer*. Lawrence Erlbaum Publishers, Hillsdale NJ, 1989.

[SPL*85] J.C. Spohrer, E.P ope, M. Lipman, W. Sack, S. Freiman, D. Littman, W.L.Johnson, and E. Soloway. *BUG CATALOGUE: II,III, IV*. CS TR 386, Yale New Haven CT, May 1985.

[S89] J.C. Spohrer. MARCEL: A GTD impasse/repair model of student program generation and individual differences. Ph.D. Diss. , Yale University, T.R. 687, 1989

Rule Induction and Interference in the Absence of Feedback: A Classifier System Model

Lewis G. Roussel, Robert C. Mathews, and Barry B. Druhan

Louisiana State University

Department of Psychology

ABSTRACT

This study extends the work of Druhan et al. (1989) and Mathews et al. (1989b) by applying their computational model of implicit learning to the task of learning artificial grammars (AG) without feedback. The ability of two induction algorithms, the forgetting algorithm which learns by inducing new rules from presented exemplars and the genetic algorithm which heuristically explores the space of possible rules, to induce the grammar rules through experience with exemplars of the grammar is evaluated and compared with data collected from human subjects performing the same AG task. The computational model, based on Holland et al.'s (1986) induction theory represents knowledge about the grammar as a set of partially valid condition-action rules that compete for control of response selection. The induction algorithms induce new rules that enter into competition with existing rules. The strengths of rules are modified by internally generated feedback. Strength accrues to those rules that best represent the structure present in the presented exemplars. We hypothesized that the forgetting algorithm would successfully learn to discriminate valid from invalid exemplars when the set of exemplars was high in family resemblance. We also proposed that the genetic algorithm would perform better than chance but not as well as the forgetting algorithm. Results supported those hypotheses. Interestingly, the Mathews et al. (1989a) subjects performed no better than chance on the same AG learning task. We concluded that this discrepancy between the simulation results and the human data is caused by interference from unconstrained hypotheses generation of our human subjects. Support for this conclusion is two-fold: (1) subjects are able to learn the AG when the task is designed so that hypothesis generation is inhibited, and (2) informal inspection of verbal protocols from human subjects indicates they are generating and maintaining hypotheses of little or no validity.

INTRODUCTION

Some models have been proposed to account for learning in situations where external feedback is either missing or unreliable (eg., Fried & Holyoak, 1984; Billman & Heit, 1988). Generally, these models work by detecting environmental regularity in the form of either frequency cues or feature covariation. Models of implicit learning, which also operate by detecting and encoding structural features in the input, would seem to be perfectly capable of learning in such an environment. A model (THIYOS) based on a classifier system has been found useful for modeling implicit learning of artificial grammars when external feedback is present (Mathews, Druhan, & Roussel, 1989). We are encouraged by this result because the induction model of Holland, Holyoak, Nisbett, and Thagard (1986) from which THIYOS is derived is claimed to be relevant to a wide range of task domains, from conditioning to stereotyping. This paper will present evidence that THIYOS can learn in the absence of external feedback.

Recently Mathews et al. (1989b) applied three different induction algorithms to learning artificial grammars (AG) with THIYOS. The AG task consisted of a series of trials in which THIYOS attempted to select the valid string from a list of five alternatives and was given feedback about the correct choice after each trial. On each trial four of the strings contained from one to four violations (incorrect letters). The simulation data suggested that the effectiveness of the algorithm depends on how data

driven the task is. One of the algorithms, termed "forgetting", was shown to be superior for learning an AG in this multiple choice task because it best exploits clues to the underlying rule structure provided by new exemplars across trials.

Although the Mathews et al. (1989b) data were collected under conditions of continues feedback about correctness of choices, we hypothesized that the forgetting algorithm would operate well in such a data driven task even without feedback. Given that all of the choices contain grammatical features (e.g., valid bigrams or trigrams), even when wrong choices are selected valid rules could be generated by the forgetting algorithm. Further, valid rules should win out across trials over less valid rules because valid features will apply more often in the entire population of choices. To test this hypothesis, THIYOS was modified to operate without feedback and then applied to the task of learning the same two artificial grammars studied previously by Mathews et al. (1989b). After a brief description of the learning task and THIYOS, the experiment and results will be discussed along with some comparisons of simulation data with data collected from human subjects who also tried to learn the grammars without feedback.

THE LEARNING TASK

On each trial a set of five letter-strings were presented. The strings on each trial consisted of one completely valid string, one string with one letter that could not occur in a particular position, one string with two wrong letters, one with three, and one with four invalid letters. The object was to select which of the five choices was the correct string. No feedback about the correctness of the choice was given. This procedure continued for 200 trials and was repeated three times with three sets of stimulus items. Data from human subjects learning the grammar under identical conditions were collected in a previous study (Mathews, Buss, Stanley, Blanchard-Fields, Cho, and Druhan, 1989). Subjects in that study participated in three sessions distributed over a three week period. Each session consisted of 200 trials. The stimulus items used in the simulations were identical to those in the Mathews, et al. study. The initial block of ten trials each week contained all new items and then each successive block of trials contained five old items repeated from previous trials that week and five new items. In sessions 2 and 3, most items (old or new) occurred in previous sessions.

Both grammars used in the Mathews, et al. (1989a) study were used in this study. The finite state grammar is illustrated in figure 1a. A valid string is any sequence of letters generated by following a pattern of arrows from left to right. Strings generated by this grammar tend to share many features, that is they have a high level of family resemblance among exemplars. The biconditional grammar is illustrated in figure 1b. It contains three letter association rules that determine what pairs of letters must occur in corresponding positions in the first and second halves of the string. As illustrated in figure 1b, the three association rules are S goes with V, C goes with P, and T goes with X. The first four letters in a valid string can be any combination of the six letters, but once they are selected, they completely determine what the second set of four letters must be (see figure 1b). Exemplars of the biconditional grammar do not share any common sequences of initial or final letter patterns and, therefore, they have a lower level of family resemblance among exemplars.

THE MODEL

The computational model described here is essentially a classifier system model with certain modifications that make it amenable to modeling artificial grammars in general and the particular task specifically (see Holland et al., 1986). Classifier systems are a

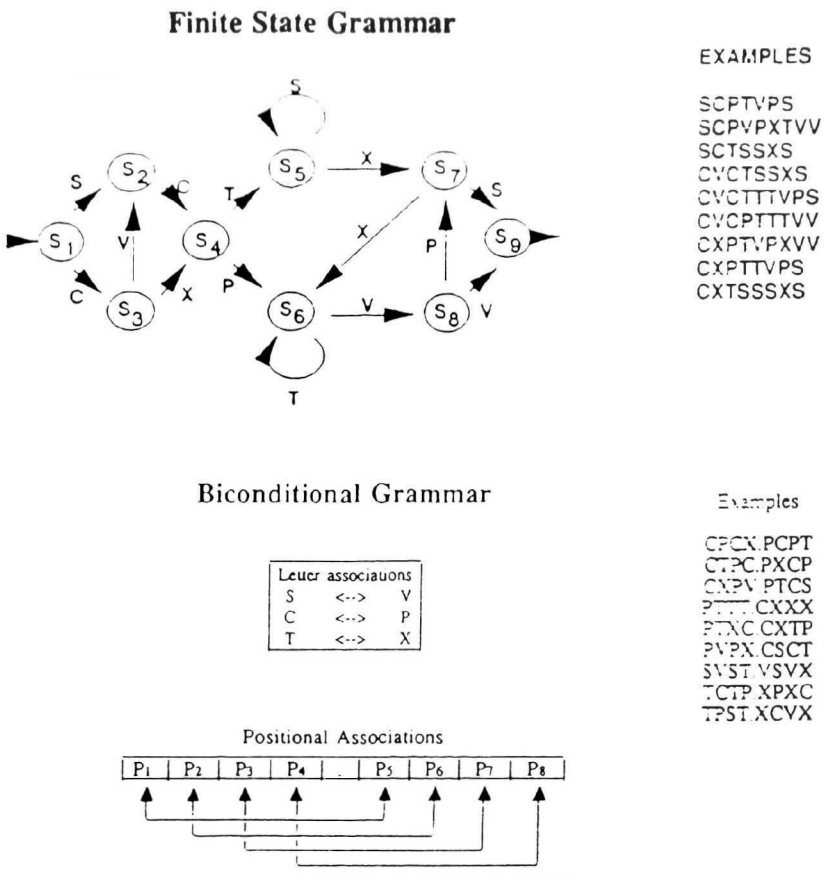


Figure 1a and 1b.

type of production system model with some specific processing assumptions. First, the productions, condition-action pairs, are composed of strings of equal length. Each condition-action pair is called simply a "classifier". Each element can be thought of as representing a unique feature of the object or event being described by the classifier. Complex objects or events can be coded by adding conditions to the condition side of the classifier--each classifier has only one action.

Rules are represented as condition-action pairs, in which both the conditions and the actions are fixed length strings of letters, numbers, "#"s, and "_"s. The exemplars of the grammar to be learned are represented in a similar fashion such that the lengths of the string, action string, and the exemplar string are all equal. In order to determine whether a rule applies, its condition side is matched position by position against the exemplar string. The "#"s are a sort of wildcard character that will match anything. In addition, the "#"s act as variables in that they can pass information through from a message to an action. Consider the following example: the rule "if the string begins with SCT then choose it" would be represented in THIYOS as:

"##SCT########0###00 I 02CHOOSE_____######".
condition action

The five alternative strings are placed on the message list in a similar format. For example, the above rule would match an exemplar on the message list such as: "01SCTVPXVV___#10###". Numbers at the beginning of the strings are tags which differentiate strings coming from the input interface from those going to the output

interface. In the exemplar string, the "1" and "0" in the 14th and 15th positions indicate that it is choice number 1 for the given trial, and that it has zero violations. Since the corresponding positions in the condition and action side of the classifier contain "#"'s, the "1" and "0" are passed through from the exemplar to the action. Since the action of this classifier is tagged for the output interface, it would tell the system to choose letter string number 1. The execution cycle performs one trial per cycle by iterating through the following steps:

1) Read in the five alternative exemplars from the input interface, and place them on the message list.

2) Compare the condition sides of all rules to each message on the message list and record all matches.

3) Calculate a bid for each classifier on the interim list using the parameters of strength, specificity, and support.

4) Select the w highest bidders and allow these classifiers to post their messages on an interim message list. (The size of the set (w) reflects the model's assumptions about working memory limitations.)

5) Recalculate the bids for all classifiers on the interim list and post their messages on a new message list. This rebidding is necessary because some classifiers lose the support of classifiers that failed to make it to the interim list.

6) All rules on the new message list loose a portion of their strength as payout for the privilege of posting their messages.

7) Process the contents of the new message list through an output interface which strips off messages tagged for output. The highest bidder and all rules whose messages agree with the highest bidder are rewarded by incrementing their strength.

8) Depending on which learning algorithm is being applied, a new rule(s) is created by that algorithm and added to the system.

9) Replace the old message list with the new one.

10) Reduce the strength of all rules in the system. This step simulates the forgetting of rules which seldom or never compete and apply.

11) Return to step 1.

This process continues until all trials have been completed.

Within the Induction framework of Holland, et al. (1986) an algorithm termed the "bucket brigade" is used to allocate activation or strength to good rules in the system across different time steps based on strength, specificity, and support. A similar mechanism is used here with some necessary modifications. In the current model each trial is processed in discrete fashion. No chaining between rules takes place (e.g., see Holland et al., 1986). Therefore the calculation of support was modified to represent the level of agreement among rules on a single time-step rather than the relevance of a rule to the context of the previous time-step (see Druhan & Mathews, 1989, for a more detailed discussion). Strength is a numerical measure of the level of success a rule has had in representing the environment. Specificity is a measure of how complete that representation is (Holland et al., 1986). The learning algorithms

We examined two algorithms for creating rules. The first, the genetic algorithm, is inspired from genetics and involves mutation and crossover of different parts of existing rules. The second, the forgetting algorithm, makes new rules from presented

exemplars by retaining a subset of features of that item and creating a new rule that says to select strings with those features.

The genetic algorithm. The genetic algorithm uses two methods or "operators" for altering existing rules to make new rules. The first, crossover, splits pairs of rules into two parts and recombines the parts to make new rules. Candidates for crossover are selected from those rules posting messages on the current trial. These candidates are copies, ranked by strength and paired off so that the strongest two go together, and the next stronges pair, etc.. Then a random split point is selected in each pair of classifiers and the copies are split and recombined by exchanging their initial and final portions. Thus two new rules are created from each pair of rules and the original or parent rules are left intact in the set of rules.

The second genetic operator, mutation, creates a copy of a randomly selected original rule and randomly changes one character, making a new rule. The mutation rate is set very low in this simulation; each existing rule in the rule set mutates with a probability of 0.001. The purpose of mutation is to keep some "new blood" in the rule system. Without mutation, crossover might tend to inbreed certain types of rules and effectively lock out discovery of radical different rules.

The genetic algorithm is best characterized as top-down or conceptually driven. Its search through the space of possible rules is guided mainly by the knowledge contained in its existing rule set. Input serves only to qualify the goodness of that knowledge by strengthening those rules created by the algorithm that have posted their messages. Therefore we hypothesized that, in the absence of external feedback, the genetic algorithm would eventually succeed in inducing a representation of the grammar but would proceed slowly.

The forgetting algorithm. The forgetting algorithm extracts features from the choice selected by the simulation on the current trial and makes it a new rule. In this algorithm features of the choice are included in the new rule probabilistically. A serial position curve is used to set the probabilities of incorporating features into the new rule.

One interesting aspect of the forgetting algorithm concerns the tradeoff between remembering too much versus too little of an exemplar. If all the features of the exemplar were incorporated into the new rule it could only be applied to that one exemplar. On the other hand remembering too little about a past exemplar and creating a new rule based on only one feature of the exemplar would tend to produce rules that have little or no validity. Thus, it is better to forget part of the exemplars one sees in creating new rules--hence the name forgetting algorithm.

The forgetting algorithm is data driven in that it induces rules based solely on the input. Because all items to choose from are either valid or distortions of valid items, all share features in common with valid strings across trials and, consequently, whatever structure that exists in the input will be captured in the set of induced rules. This suggests that "forgetting" should perform well in the present task even though external feedback is absent.

THE SIMULATION

The two algorithms were applied to the two grammars, finite state and biconditional. The genetic algorithm needs rules to "mate" and produce "offspring" so a set of initial rules was constructed and entered into THIYOS at the start of each run. This initial set

of rules represents the basic knowledge that any subject would bring to the task. This knowledge consists of the set of all possible rules for string selection based on single letters in a specific position (e.g., "choose strings that have an "S" in position 3"). Runs of the forgetting algorithm were initiated with the same set of initial rules.

RESULTS

The dependent variable is the number of errors (incorrect strings) selected in trials 11 through 200 of each session (trials 1 through 10 were not included in the analyses because the distribution of old versus new items in those trials differed from all other trials). Since each trial consisted of five choices, there is a 0.80 probability of committing an error by chance. Therefore, chance performance is 152 errors per session.

The mean performance for each condition is plotted in figure 2. In an analysis of variance (ANOVA) of the error data the overall group (algorithm) effect was significant for both grammars with $F(1,18)=56.26$, $p<0.0001$ and $F(1,18)=11.23$, $p<0.0036$ for the finite state grammar and the biconditional respectively. For the finite state grammar the forgetting algorithm performed better than genetic. Both were significantly better than chance by session three. For the biconditional grammar, the same pattern, forgetting better than genetic, was obtained. Again, both were significantly better than chance by

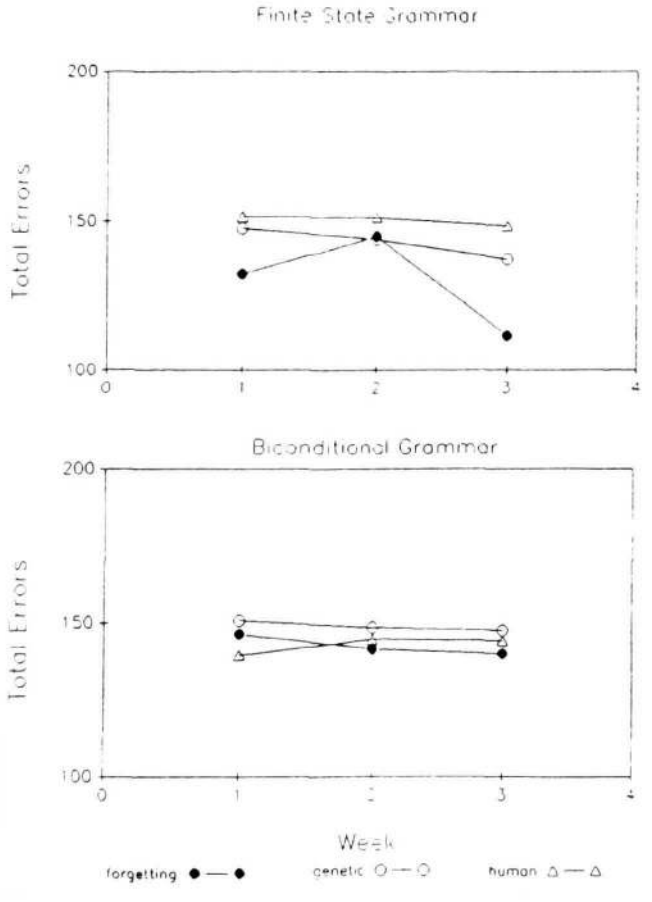


Figure 2.

session three.

Data from Mathews, et al. (1989a, experiment 1) were included in a second analysis to determine how well the simulation fit the human data. Subjects had been trained with

the same stimulus items used in the simulation. These subjects had performed the same multiple choice task under no-feedback conditions. The human subjects performed at chance for all three sessions and for both grammars (see figure 2).

DISCUSSION

Two general conclusions can be drawn. First, THIYOS combined with either of the two learning algorithms studied was able to perform better than chance in the grammar learning task even though the simulation received no feedback. This supports our hypothesis that the forgetting algorithm should be able to operate in the absence of feedback. The genetic algorithm was also successful but always performed less well than forgetting.

The second conclusion to be drawn concerns the performance of our human subjects: they were unable to learn about these artificial grammars when feedback was absent. This is a puzzle about which one can intelligently speculate in the light of the results from other artificial grammar (AG) learning experiments.

A tentative explanation for this discrepancy between the simulation results and data from human subjects would be that the performance of human subjects is hampered because their explicit, conscious strategies interfere with implicit learning. That interference is responsible for the poor performance of the subjects, is supported by several studies. Reber and his colleagues (e.g., Reber, 1976; Reber, Kassin, Cantor, & Lewis, 1980) demonstrated that subjects performed poorly in an AG learning task when given instructions to search for rules. He attributed this interference to explicit processing mechanisms. In the Reber et al. (1980) study, presenting the stimuli in a structured display that revealed the family resemblance of the items resulted in better performance with explicit processing whereas presenting the items in a random fashion (unstructured condition) resulted in better performance with implicit processing. The present multiple choice task is similar to the unstructured condition because the five choices on a particular trial are selected randomly and share little family resemblance. Therefore it is reasonable to expect that explicit processes could interfere with learning in the present task.

On going work in our laboratory using a different paradigm for studying implicit learning indicates that subjects can acquire knowledge about the Mathews et al. finite state grammar without feedback during training. The paradigm is based on the Hebb effect. Subjects were presented letter strings, one per trial, and asked to hold each in memory while performing a distractor task. After the distractor task they attempted to recall as many letters of the string as possible. The subjects were not told that every third string was a valid string while all others were random strings. No feedback was given. Results indicate that subjects improved in recall of valid strings relative to recall of random strings across trials (showing a Hebb effect on valid strings) and they subsequently performed above chance on a string discrimination task. Thus humans can learn the finite state grammar without feedback when they are not explicitly forming hypotheses during learning.

Finally, when feedback is absent, there is little to guide the hypothesis generating processes toward solution. More importantly, without feedback, the hypothesis generating mechanism is virtually unrestrained. Completely wrong hypotheses can be generated and strengthened across trials. Verbal protocols of our human subjects taken after the experiment support this conclusion. Typically, subjects generated invalid rules such as "pick strings that look like words" or pick strings that have letters lower in the alphabet". We are currently testing the notion that such invalid rules can block

the implicit learning mechanism by introducting one or more of their bad rules in a THIYOS simulation. We predict that their rules may gain strength and prevent the generation of valid rules.

If such simple mechanisms as the forgetting algorithm are capable of exploiting structured stimulus domains without error correcting feedback, we should expect that living systems would have evolved similar capacities. The fact that the process was completely stopped by unrestrained hypothesis testing in our human subjects is quite interesting and it adds further impetus to the postulation of two distinct learning mechanisms.

REFERENCES

Billman, D, & Heitman, E. (1988). Observational learning from internal feedback: A simulation of an adaptive learning method. *Cognitive Science, 12,* 587-625.

Druhan, B. B., & Mathews, R. C. (1989). THIYOS: A classifier system model of implicit knowledge of artificial grammars. *Paper presented at the annual meeting of the Cognitive Science Society, Ann Arbor.*

Fried, L. S., & Holyoak, K. J. (1984). Induction of category distributions: A framework for classification learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 10,* 234-257.

Holland, J. H., Holyoak, K. J., Nisbett, R. E., & Thagard, P. R. (1986). *Induction: Processes of inference, learning and discovery.* Cambridge, MA: The MIT Press.

Mathews, R. C., Buss, R. R., Stanley, W. B., Blanchard-Fields, F., Cho, J. R., & Druhan, B. (1989). Role of implicit and explicit processes in learning from examples: A synergistic effect. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15,* 1083-1100.

Mathews, R. C., Druhan, B. B., & Roussel, L. G. (1989). Forgetting is learning: Evaluation of three induction algorithms for learning artificial grammars. *Paper presented at the annual meeting of The Psychonomic Society, Atlanta.*

Reber, A. S. (1976). Implicit learning of synthetic languages: The role of instructional set. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 2,* 88-94.

Reber, A. S., Kassin, S. M., Lewis, S., & Cantor, G. (1980). On the relationship between implicit and explicit modes in the learning of a complex rule structure. *Journal of Experimental Psychology: Human Learning and Memory, 6,* 492-502.

ACKNOWLEDGEMENTS

This research was supported in part by National Science Foundation Grant #BNS-8509493 to Robert C. Mathews, Ray R. Buss, and William B. Stanley, and in part by the Louisiana Board of Regents Grant #86-LBR(21)- 021-10 through the Louisiana Quality Education Support Fund to Robert C. Mathews.

ASSOCIATIVE MEMORY-BASED REASONING: SOME EXPERIMENTAL RESULTS

Boicho Nikolov Kokinov

Institute of Mathematics
Bulgarian Academy of Sciences
Bl.8, Acad. G. Bonchev Street
Sofia 1113, Bulgaria

ABSTRACT

Deduction, induction and analogy are considered as slightly different manifestations of one and the same reasoning process. A model of this reasoning process called associative memory-based reasoning is proposed. A computer simulation demonstrates that deduction, induction and analogy in problem solving could be performed by a single mechanism which combines the neural network approach with symbol level processing. Psychological experiments on priming effects in problem solving tasks have been carried out in order to test the hypothesis about the uniformity of human reasoning. In particular it has been shown that there are priming effects in all three cases (deduction, induction and analogy) and these priming effects decrease in the course of time which corresponds to the model's predictions based on the retrieval mechanism. The computer simulation demonstrates the same type of priming effects as observed in the psychological experiments.

1. INTRODUCTION.

The tradition of exploring deduction, induction and analogy in AI is to study them separately and to propose different models for them. Contrary to that we believe that there is a uniform underlying reasoning process of which deduction, induction and analogy are only slightly different manifestations. A model of this process in the context of problem solving has been proposed in [11]. This model has been called associative memory-based reasoning (AMBR).

The associative memory-based reasoning model will be reviewed in section 2. The retrieval step will be discussed in more detail.

Psychological experiments have been carried out in order to explore the common features of deduction, induction and analogy and thus to test the hypothesis about the uniformity of human reasoning. The experiments also test the model's predictions about the results of the retrieval step. The experiments will be described in section 3.

A computer implementation of associative memory-based reasoning has been developed. The results from the computer simulation are discussed in section 4.

2. ASSOCIATIVE MEMORY-BASED REASONING: An Overview

AMBR was proposed in [11]. It is composed of a set of processes running sequentially or in parallel: *retrieving*, *mapping*, *transferring*, *evaluation* and *updating processes*. Depending on the results of the retrieval process and the

correspondence between the descriptions established during the mapping process, we can view the reasoning process as deduction, induction or analogy. Burstein and Collins [3] similarly conclude, after analyzing a large set of protocols, that "the kind of knowledge retrieved from memory drives the particular line of inference produced.".

AMBR is based on a hybrid architecture put forth in [10,12]. This architecture of cognition combines a frame-like representation scheme with a spreading activation mechanism (called in our model associative mechanism) implemented as a neural network. In this way we exploit the advantages of theories both at the symbol and subsymbol levels. In contrast with Anderson's ACT* model [1] we do not separate declarative from procedural knowledge and, since a frame-like representation scheme is used, the domain of application of spreading activation is extended over procedural knowledge as well. Both types of spreading activation, automatic and directed, exist in AMBR versus automatic only in ACT*. Finally, reasoning in ACT* is rule-based whereas rules and other general knowledge as well as particular case descriptions are used in AMBR.

Unlike case-based reasoning [4,5,14] (with a sequential retrieval process relying on a static indexing mechanism), in AMBR we have a parallel and continuous retrieval process depending both on the static links and the current memory state. Besides previous cases, AMBR uses general knowledge about a class of problems and general algorithms as well.

AMBR differs from memory-based reasoning (Stanfill and Waltz, [17]) in that it relies on the availability of general knowledge and domain models in memory. It uses a sophisticated frame-like representation and a complex mapping process.

2.1. Memory Organization

Our memory model includes *long term memory (LTM)*, *working memory (WM)* and *focus* (of attention), where WM is the active part of LTM, and the focus is the most active element of WM. LTM is considered as a network of nodes and weighted links. The nodes correspond to frame-like descriptions of objects, concepts, events or plans. The links correspond to semantic relations as well as to arbitrary associations. The nodes have variable levels of activation. All nodes whose level of activation is above their threshold t_i form WM. The focus is the only node which can be consciously processed. A more detailed description of the representation scheme and memory organization can be found in [12].

2.2. Associative Mechanism

The associative mechanism runs in parallel with all other processes and in this way influence their work. It changes the activity of the nodes which is critical for all processes working on these nodes. The associative mechanism plays the role of a retrieval mechanism as well. It can be considered as a form of spreading activation [1,2]. Its detailed description in the terms of PDP models [16] can be found in [13]. There are two slightly different versions of the associative mechanism: directed and automatic.

Directed spreading activation is performed when some information has to be obtained from a node referred to by the focus. This is done by directing the activation via the corresponding semantic link. It usually has the maximum strength of 1 and therefore the node pointed to by the link will receive the same activation as the focus and thus become the new focus.

Automatic spreading activation is performed continuously and in parallel with all other processes in memory. Each node in working memory passes over its

activity to all of its neighbours proportionally to the strength of the corresponding link.

2.3. Reasoning in Problem Solving

AMBR is started when a problem description is in the focus. The retrieval process ends with a new memory state, i.e. a new collection of nodes in WM and a new focus. Then a mapping process between the old (O) and the new (N) focus is started. The mapping aims to establish a correspondence between the slots of the problem description (O) and the retrieved one (N). The result from the mapping between O and N heavily depends on the activity of the other nodes in WM. The established mapping is extended by the transferring process and tested by the evaluation process, i.e. some knowledge from O is transferred and eventually applied in the new situation N.

Now we shall concentrate only on the retrieval step which is performed by means of the associative mechanism described above. The results of the retrieval step will depend on the input (problem description and concrete wording) as well as on the memory state, i.e. which nodes are active at the moment (as a result of previous input or reasoning processes). The prediction made is that there has to be a priming effect on reasoning in problem solving task and that this priming activity has to decrease in the course of time.

3. PSYCHOLOGICAL EXPERIMENTS

Psychological experiments are performed in order to test our two main hypotheses:

- *first*, that deduction, induction and analogy are performed by a single uniform reasoning mechanism;

- *second*, that the retrieval step in this reasoning process is accomplished by a spreading activation mechanism.

The spreading activation hypothesis is usually tested by priming experiments. Most of the experiments of this kind are performed with low level tasks (lexical decision, items recognition, word completion, etc.). We have shown priming effects in a question-answering (fact retrieval) task with an experiment described in [12].

The experiments to be described here are carried out in order to explore the existence of ***priming effects in a high level task like problem solving.*** Gick and Holyoak [7,8] have performed some experiments on analogical problem solving, but they concentrate more on the mapping process rather than on the retrieving one (the subjects usually received hints to apply a given analogy to the problem). But as Holland, Holyoak et. al.[9] pointed out, "for an autonomous problem solver the most difficult step in the use of analogy is likely to be the retrieval of a plausible source".

According to our second hypothesis, we expect that the activation of some specific knowledge before giving the problem to the subjects will influence the way of solving the problem or the success/failure ratio. Moreover, we expect this priming effect to decrease in the course of time.

Considering the first hypothesis, we expect that the same priming effect should be observed in all three cases: deduction (Experiment I), induction (Experiment II) and reasoning by analogy (Experiment III).

3.1. Experiment I.
Priming Effects on Deductive Reasoning

3.1.1. Method

Subjects. The subjects were 385 pupils from 15 to 17 years old. They participated in the experiment without pay.

Materials. The subjects had to solve mathematical problems that belonged to well-known classes, applying well-known general methods in a deductive manner.

The target problem was: "Prove the identity: $\sqrt{7 + \sqrt{24}} - \sqrt{7 - \sqrt{24}} = 2$ "

The priming problem was: "Reduce the expression $\sqrt{6 + \sqrt{20}}$ "

The distractor problems were ones from geometry.

The target problem could be solved in at least two ways, well-known by the pupils beforehand:

- squaring both sides of the identity
- reducing the expressions under the radicals,
  e.g. $7 + \sqrt{24} = 1 + 6 + 2\sqrt{6} = (1 + \sqrt{6})^2$

We suppose that the priming problem activates the knowledge needed to solve the target problem in the second way.

Procedure. Subjects were tested in groups of 15 to 25 people. They had to solve about 10 problems, each of them in a restricted time period. The problem descriptions were written on a blackboard one by one - after the time for solving a given problem was up, the blackboard was cleaned and the next problem was written on it. Subjects were not aware of our classification of problems as target, priming and distractor ones. The experimenter presented the solution of the priming problem to the subjects in order to ensure that all of them will use one and the same method and thus activate the same general knowledge. The solutions of some of the distractor problems were also presented to the subjects so that the priming problem would not stand out among the rest. Subjects wrote down the solutions of the problems keeping trace of all attempts and errors. The statistics made in this experiment reflects the number of subjects trying to solve the problem in one and the same way but does not reflect the number of correct solutions of the problems.

Design. There were three conditions in the experiment:

- control condition - subjects solved only distractor and target problems;
- near priming condition - subjects solved the priming problem immediately before the target one;
- far priming condition - subjects solved the priming problem, after that two distractor problems (8 min) and then the target problem.

The measured variable is the number of subjects solving the target problem in each particular way.

3.1.2. Results.

The number of subjects that solved the target problem in a certain way under each condition is given in Table 1. The differences between the results in every two conditions are significant at the 1% level measured by the X^2 (chi square) criterion. Although almost all subjects have solved the target problem "by squaring" under the control condition, the experiment demonstrates a clear priming effect which decreases slowly. There are both near and far priming effects but they are significantly different.

ways\conditions	control	near prime	far prime
solved by squaring	71	43	54
solved by reducing	7	91	57
unable to solve	13	15	34

Table 1
Results from Experiment I.
X^2=69.04 (control - near), X^2=40.52 (control - far)
X^2=16.37 (near - far)

3.2. Experiment II.
Priming Effects on Inductive Reasoning (Generalization)

3.2.1. Method

Subjects. Subjects were 422 pupils from 15 to 17 years old participating in the experiment without pay.

Materials. Subjects had to solve about 10 problems.

The target problem was Clement's "spring coils" problem: Imagine a spring with a weight hung on it; if the original spring is replaced with one made of the same kind of wire, with the same number of coils, but with coils twice as wide in diameter, will the spring now stretch more, less, or the same amount under the same weight, and why?

The priming problem I was: There are two rods made of one and the same material with the same width and profile but with different lengths. Each rod is fixed horizontally by one of its ends and a weight (the same for both rods) is hung at the other end. Which rod will bend more?

The priming problem II was: There are two single-coil springs made of one and the same kind of wire but with different diameters of the coils. Which coil will stretch more under one and the same weight?

The distractor problems were mathematical and physical ones.

The solution of the bending rod problem is well known by pupils. Subjects have no general knowledge to solve the target problem in a deductive way, but they are able to generalize the solution of the bending rod problem to a solution of the target problem in an inductive way (i.e. the rod need not be straight). We expect that the priming problem will activate the required knowledge and in this way facilitate the solving of the target problem.

It might be questionable if the target is solved by generalization of the bending rod problem or it is used as a base for analogy. (This shows that the boundaries between analogy and generalization are not always quite distinct.) Because of the fact that the protocols from the experiment were not always clear enough to make this distinction, we repeated the experiment (with other subjects) with the priming problem II, where the target problem is obviously a generalization of the priming problem with respect of the number of coils. (According to Clement [6] this case cannot be considered as analogy.)

Procedure. The procedure is the same as in Experiment I.

Design. There were three conditions in the experiment:

- control condition - subjects solved only distractor and target problems;
- near priming condition I - subjects solved the priming problem I just before the target one;

- near priming condition II - subjects solved the priming problem II just before the target one;

- far priming condition I - subjects solved the priming problem I, after that two distractor problems (4 min) and then the target problem.

- far priming condition II - subjects solved the priming problem II, after that two distractor problems (4 min) and then the target problem.

The measured variable is the number of subjects answering the target problem in each possible way.

3.2.2. Results.

The number of subjects who answered the target problem in a certain way under each condition is given in Table 2. Both near and far priming conditions are significantly different from the control condition at the 1% level measured by the X^2 criterion.

The difference between the results under near and far priming conditions has not been found to be statistically significant but there is a tendency towards decreasing the priming effect. We think that the priming effect is not significantly decreased in this case because of the short time period between the priming and test phase (4 min).

answers	control	near I	far I	near II	far II
more	22	82	70	58	64
equal	6	3	4	5	3
less	23	30	30	6	16

Table 2
Results from Experiment II.
X^2=13.94 (control - near I), X^2=9.34 (control - far I)
X^2=24.1 (control - near II), X^2=16.04 (control - far II)
X^2=5.39 (near I - far I), X^2=4.09 (near II - far II)

3.3. Experiment III.
Priming Effects on Reasoning by Analogy

3.3.1. Method

Subjects. The subjects were 344 pupils from 15 to 17 years old participating in the experiment without pay.

Materials. Subjects had to solve about 10 problems.

The target problem was: Imagine you are in a forest by a river and you want to boil an egg. You have only a knife, an axe and a match-box. You have no containers of any kind. You could cut a vessel of wood but it would burn in the fire. How would you boil your egg?

The priming problem was: How can you make tea in a glass?

The distractor problems were mathematical and physical ones.

Subjects have no general knowledge for solving the target problem in a deductive way; they do not know any particular case of this situation either. All of them, however, have experience in using immersion heaters (i.e. they know the solution of the priming problem) and could try to find an analogy (to heat stones or the knife or the axe and to put it in the water). We expect that the

priming problem activates the knowledge about immersion heater and thus facilitates the analogy.

Procedure. The procedure was the same as in Experiment I. The experimenter presented the solution of the priming problem to the subjects in order to ensure that all of them will focus on the same particular experience (case).

Design. There were four conditions in the experiment:

- control condition - subjects solved only distractor and target problems;
- near-1 condition - subjects solved the priming problem just before the target one; the experimenter presented the solution with the words: "By an immersion heater".
- near-2 condition - subjects solved the priming problem just before the target one; the experimenter presented the solution with the words: "We heat the water by an immersion heater.".
- far condition - subjects solved the priming problem, after that eight distractor problems (24 min) and then the target problem.

The measured variable is the successes/failures ratio in solving the target problem.

3.3.2. Results

The number of subjects that successfully solved the target problem under each condition is given in Table 3. The differences between the results in control and in both near priming conditions are significant at the 5% and 1% level respectively, measured by the X^2 criterion. There are no significant differences between the control and the far condition. We think that there is no far priming effect because of the long time period between the priming and the test phase (24 min).

results\conditions	control	near-1	near-2	far
success	7	31	71	7
failure	34	54	90	50

Table 3.
Results of Experiment III.
X^2=4.94 (control - near-1), X^2=10.07 (control - near-2)
X^2=0.45 (control - far), X^2=10.18 (near-1 - far),
X^2=18.55 (near-2 - far)

3.4. Discussion

The experiments reported here have shown that **1) there is a priming effect in all three cases (deduction, induction and analogy); 2) this priming effect decreases in the course of time** (with no statistical significance in the induction case) and **3) it decreases slowly and there is still a priming effect after solving some distractor problems (for 4 to 8 min).** Experiment III has demonstrated a tendency (without statistical significance) that different wordings of the priming solution can influence the solving of the target problem differently. All these results substantiate our second hypothesis that the retrieving step is performed by a spreading activation mechanism.

Recently a retrieval theory [15] has been proposed which assumes that the prime and the target are combined at retrieval into a compound cue that is used to access memory and in this way some priming effects can be accounted for as well. We think, however, that it will be more difficult to explain the far

priming effects demonstrated in our experiments with the help of that theory. The priming effects demonstrated in low level tasks are indeed short term effects. In our opinion this is due to the fact that in these experiments only single nodes are activated by the preliminary setting (e.g. only one or two words). In more complex experiments like the ones reported here a large part of the network is activated and the activation pattern is more stable and that is why the activation of the nodes decreases more slowly (the nodes activate each other mutually).

The experimental results reported here also substantiate our first hypothesis that deduction, induction and analogy are performed by a single uniform mechanism because the effects demonstrated in all three experiments are the same.

4. COMPUTER SIMULATION

A computer implementation of associative memory-based reasoning is developed [13]. It is done in Common Lisp. It simulates human problem solving in the area used in Experiment III (cooking and boiling water, eggs, etc. in the kitchen or in the forest). The simulation system demonstrates that AMBR is capable of problem solving in a deductive, inductive and analogical way as well as some of the priming effects found in our experiments. For example, without preliminary activation of any knowledge the system failed in solving the target problem in Experiment III, but after preliminary activation of the case of preparing tea by an immersion heater in a glass, the system found a possible solution - using a hot knife, if the stone description is also activated beforehand than the generated by the system solution is "using a hot stone". A detailed description of the simulation experiments will be presented in another paper.

5. CONCLUSIONS

Two hypotheses underlying associative memory-based reasoning have been discussed and some experimental support for these hypotheses has been obtained.

Both **near priming effect** (immediately after the priming) and **far priming effect** (4 to 8 min after priming) **on problem solving in high level tasks** have been demonstrated. It has also been shown that the **priming effect decreases in the course of time.** This is considered as a support for the spreading activation hypothesis, which needs further testing. Psychological experiments are planned to explore the causes for the priming effects: we would like to contrast the hypothesis of the residual activation after the priming phase with the hypothesis of the creation of associative links during the priming phase. The decreasing of the priming effect in the course of time substantiates the first hypothesis but nevertheless we would like to explore the differences further by giving two different primes to the subjects in order to check if there will be interference between the priming effects.

The experiments on deductive, inductive and analogical problem solving have demonstrated the same behavioural phenomena which is considered as a support for the hypothesis that there is a single uniform reasoning mechanism.

The simulation results have shown that it is possible to produce deductive, inductive and analogical reasoning by the AMBR mechanism and that the demonstrated priming effects are similar to that in the performed psychological experiments.

ACKNOWLEDGEMENTS.

I am grateful to P. Barnev, S. Zlateva, E. Gerganov and to all participants in the interdisciplinary seminar on Cognitive Science at the Institute of Mathematics, B.A.S. for the valuable discussions relevant to the paper. I thank V. Nikolov, Ch. Mirchev, G. Chokoev, D. Kozaliev and all school teachers who helped organize the experiments for their assistance. This work was partially supported by Contract No 607 with the Ministry of Science and Education and by project 1001002 with BAS.

REFERENCES

[1] Anderson, J. - The Architecture of Cognition, Harvard Univ. Press, Cambridge, MA, 1983.

[2] Anderson, J. - Spreading Activation. In: Anderson, Kosslyn (eds.) - Tutorials in Learning and Memory, W. H. Freeman, San Francisco, CA, 1984.

[3] Burstein M., Collins A. - Modeling a Theory of Human Plausible Reasoning. In: T.O'Shea & V.Sgurev (eds.) Artificial Intelligence III, Elsevier Science Publ., 1988.

[4] Carbonell, J. - Learning by Analogy: Formulating and Generalizing Plans from Past Experience. In: Michalski, Carbonell, Mitchell (eds.) Mach. Learning, Tioga Publ. Comp., 1983.

[5] Carbonell, J. - Derivational Analogy: A Theory of Reconstructive Problem Solving and Experience Acquisition. In: Michalski, Carbonell, Mitchell (eds.), Mach. Learning II, Morgan-Kaufman, 1986.

[6] Clement, J. Observed Methods for Generating Analogies in Scientific Problem Solving. Cognitive Science, vol. 12, 1988

[7] Gick, M., Holyoak, K. - Analogical Problem Solving, Cognitive Psych. 12(80), 306-356, 1980.

[8] Gick, M., Holyoak, K. - Scheme Induction and Analogical Transfer, Cognitive Psych. 15(1), 1983.

[9] Holland, J., Holyoak, K., Nisbett, R., Thagard, P. R. - Induction, MIT Press, Cambridge, MA, 1986.

[10] Kokinov, B. - Computational Model of Knowledge Organization and Reasoning in Human Memory. In: Math. and Education in Math., BAS, Sofia, 1988.

[11] Kokinov, B. - Associative Memory-Based Reasoning: How to Represent and Retrieve Cases. In: T.O'Shea & V.Sgurev (eds.) Artificial Intelligence III, Elsevier Science Publ., 1988.

[12] Kokinov, B. - About Modeling Some Aspects of Human Memory. In: Klix, Streitz, Waern, Wandke (eds.) MACINTER II, North-Holland, Amsterdam, 1989.

[13] Kokinov, B., Nikolov, V. - Associative Memory-based Reasoning: A Computer Simulation. In: Plander (ed.) Artificial Intelligence and Information-Control Systems of Robots-89, North-Holland, Amsterdam, 1989.

[14] Kolodner, J., Simpson, R. - Problem Solving and Experience. In: Kolodner, Riesbeck - Experience, Memory, and Reasoning, LEA, Hillsdale, N.J., 1986.

[15] Ratcliff, R., McKoon, G. - A Retrieval Theory of Priming in Memory. Psych. Review vol.95(3), 1988.

[16] Rumelhart D., McClelland J. and PDP Research Group - Parallel Distributed Processing, vol.1: Foundations, MIT, Cambridge, MA, 1986.

[17] Stanfill, C. & Waltz, D. - Toward Memory-Based Reasoning. Comm. of ACM, vol 29(12), 1986.

Uniformity of Associative Impairment in Amnesia

Gregory V. Jones and Siobhan B. G. MacAndrew
Department of Psychology, University of Warwick,
Coventry CV4 7AL, England

Abstract

The fragmentation model of associative memory has the attraction of specifying neither a spatial metaphor nor a symbolic representation for remembering. It was used in order to compare the recall of groups of unrelated words by amnesic and normal people. Similarly, a schema model was used in order to compare their recall of groups of related words. It was found that the impairment in remembering with amnesia revealed by these models was remarkably uniform rather than selective. This suggests that the level at which the memory storage system is damaged in amnesia is a relatively low one. In a connectionist formulation, this would presumably correspond to widespread random damage to units and the connections between them.

Some years ago, a fragmentation model of associative memory was articulated (Jones, 1976). The model was extremely simple. It proposed that the memory for an event is represented by a fragment or group of elements of that event, and that the contents of the fragment can be recalled if and only if one of this group of elements recurs as a cue. The model had some attractive features. It was parsimonious, and it gave quantitative predictions which were surprisingly accurate. However, it also had what at the time seemed to be two problems. First, it was not expressed in terms of any of the spatially localised metaphors that were then almost ubiquitous in theories of memory (Roediger, 1980). Second, it was not expressed in terms of operations upon symbolic representations in the way that, say, the state-of-the-art HAM (Anderson & Bower, 1973) and ACT (Anderson, 1976) models of memory were. The first factor could be countered by the observation of Schacter, Eich, and Tulving (1978) that the fragmentation model resembled an earlier nonlocalised theory, the resonance model of Richard Semon (1921). And the second factor could be countered by the observation of Anderson and Bower (1980) that on occasion at least the fragmentation model's predictions were more accurate than those of HAM. Nevertheless,

these two factors together perhaps meant that the fragmentation model was a bit difficult to assimilate to the then prevailing memory-model schema. We are reminded of this now because the contemporary connectionist zeitgeist has eliminated both problems. Neither spatial localisation nor symbolic representation appear as indispensable as before. With the advent of parallel distributed processing models (e.g., Hinton & Anderson, 1989; Rumelhart & McClelland, 1986) the emphasis has shifted to constructing new types of representation. These are not localised but instead distributed over many simple units, and they are also subsymbolic in that apparently complex higher order processes emerge from the iterative operation of many simple processes. So with this conceptual rehabilitation of the general nature of the fragmentation model, we decided to employ it as a tool in exploring the detailed nature of the memory deficit in organic amnesia.

To this end, we built upon previous evidence that the fragmentation model describes well the patterns of recall that occur when a person is remembering an event whose different elements are independent of each other, such as a group of words that have been selected at random (Jones, 1984). Rubin and Wallace (1989) have in fact argued convincingly that the crucial characteristic of such events is that the particular grouping should be unique. In contrast, the patterns of recall that occur when a person is remembering a group of words with a common theme are better described by a schema model (Ross & Bower, 1981). Both fragment and schema models are mathematically parametrized. We thus decided to estimate fragment and schema parameters from the remembering of unrelated and related words, respectively, and to examine the impairments in parameter values that we expected to find in moving from a normal population to an amnesic one. Let us first describe the parametrizing.

Consider a triad or group of three words. If we treat each word as equivalent, then there are three types of fragment that need to be distinguished. With probabilities ϕ, π, and $(1 - \phi - \pi)$ there may exist either a full, partial, or null fragment, respectively. A full fragment has all three words associated, a partial fragment has only two words associated, and a null fragment has no words associated. The schema model also has two parameters. The probability of one word accessing a schema is α, and that of a second word being retrieved from the schema is ρ. How do these parameters translate into predictions?

Suppose that a person is shown a triad, and then subsequently one of its words is re-presented as a retrieval cue. Further, if the first cue is unsuccessful in producing any recall, a

second word may then also be given as cue. In all, four patterns of recall may be distinguished. Two words may be recalled on the first cuing; one word may be recalled on the first cuing; no words may be recalled on the first cuing, but one word may be recalled on the second cuing; or no word may be recalled on either cuing. Let the probabilities of occurrence of these patterns be denoted by p_{20}, p_{10}, p_{01}, and p_{00}, respectively. Then the fragment predictions are that

$$p_{20} = \phi$$

$$p_{10} = 2\pi/3$$

$$p_{01} = \pi/3$$

and

$$p_{00} = (1 - \phi - \pi).$$

Similarly, the schema predictions are that

$$p_{20} = \alpha\rho^2$$

$$p_{10} = 2\alpha\rho(1 - \rho)$$

$$p_{01} = \alpha\rho(1 - \alpha)$$

and

$$p_{00} = \alpha(1 - \rho^2) + (1 - \alpha)(1 - \alpha\rho).$$

Empirical estimates of the fragment and schema parameter values may be obtained from observed frequencies of each of the four patterns or recall (denoted by n_{20}, n_{10}, n_{01}, and n_{00}, respectively) by maximizing the likelihood function

$$L = \prod_i p_i^{n_i}$$

(where i = 20, 10, 01, 00). For the fragment model, this yields the maximum-likelihood estimates

$$\phi^* = n_{20}/n$$

and

$$\pi^* = (n_{10} + n_{01})/n.$$

For the schema model, the maximum-likelihood estimates may be obtained numerically rather than algebraically.

Empirical Estimates for Amnesic and Normal Groups

An experiment of the type previously indicated was carried out, in which triads of words were presented and then cued incrementally by two of the words. For the schema estimates, the words of each triad were related via a common theme (e.g., towel, pool, goggles). For the fragment estimates, the words of each triad were unrelated (e.g., towel, convent, dune). Both fragment and schema estimates were obtained individually for each of six amnesic patients and each of seventeen normal people. The amnesic group comprised two sufferers from Korsakoff's syndrome, two people who had suffered anterior communicating

artery aneurisms, one person who had suffered encephalitis, and one person who had suffered severe frontal and temporal lobe damage in a road accident. The normal group consisted of students at Warwick University.

Two sets of 84 triads each were prepared. For one set, the words of each triad were related by a common theme. The other set was obtained by random permutation so that the three words were not related. Each person was tested half with related triads and half with unrelated triads, with the halves counterbalanced over people. In order to reduce the difference in levels of recall between the amnesic and normal groups, the former were presented with twelve sets of seven triads for 8 seconds per triad, while the latter were presented with six sets of fourteen triads for 4 sec per triad. Following the presentation of each set, each of its triads was tested by incremental cuing using two of its component words.

The frequency of occurrence of each pattern of recall in this study is shown in Table 1. It can be seen that, as expected, normal performance was better than amnesic performance, and also that related performance was better than unrelated performance. Further, if we aggregate all patterns on which some recall occurred (i.e., $[n_{20} + n_{10} + n_{01}]$ or $[42 - n_{00}]$), then we find that the amnesic decrement in performance was approximately equal for the related and for the unrelated words.

Table 1. Frequency (n_i) of each pattern of recall

		Pattern of recall			
Group	Words	20	10	01	00
Normal	Related	26.06	9.71	2.06	4.18
Amnesic	Related	4.50	7.17	1.17	29.17
Normal	Unrelated	13.41	8.94	3.59	16.06
Amnesic	Unrelated	0.83	1.83	0.33	39.00

Maximum-likelihood estimates of the schema parameters (α^* and ρ^*) and of the fragment parameters (ϕ^* and π^*) were obtained for each person from the unrelated-words data and the related-words data, respectively; the schema parameter values were obtained using a FORTRAN program employing subroutine E04JAF (minimisation subject to boundary constraints) of the NAG

library (Numerical Algorithms Group, 1977). The mean values of the schema and fragment parameters are shown in Table 2.

Table 2. Mean parameter values

Model	Parameter	Normal	Amnesic
Schema	α^*	0.861	0.491
Schema	ρ^*	0.817	0.488
Fragment	ϕ^*	0.319	0.020
Fragment	π^*	0.298	0.052

It can be seen that for the schema model both α^* and ρ^* were impaired to approximately the same extent by amnesia; similarly, for the fragment model both ϕ^* and π^* were also impaired to approximately the same extent by amnesia. A curious additional result, which seems merely coincidental rather than meaningful, was that both for the normal group and for the amnesic group the values of the α^* and ρ^* parameters were approximately equal, as were also the values of the ϕ^* and π^* parameters.

Implications

The fragment model of Jones (1976) and the schema model of Ross and Bower (1981) proved to be useful analytical tools in investigating recall impairment with amnesia. The amnesic patients were equally impaired in retaining full and in retaining partial fragments of groups of unrelated words. Similarly, they were equally impaired in accessing schemata and in retrieving from them. The two models therefore provide a description of amnesia in which recall is seen to suffer blanket impairment. This suggests that amnesia is the consequence of damage at some basic, undifferentiated level of representation. In a connectionist formulation, this would presumably correspond to widespread random damage to units and the connections between them. It should be noted, however, that although recall of related words suffered a decrement in performance with amnesia approximately equal to that for unrelated words, the absolute level of recall of related words in the amnesic group (as in the normal group) was

higher than that of unrelated words. In this respect, the results were similar to those in recent studies where priming effects have been reported with related but not with unrelated pairs of words (e.g., Schacter & Graf, 1986; Shimamura & Squire, 1984, 1989).

References

Anderson, J. R. (1976). *Language, memory, and thought.* Hillsdale, NJ: Erlbaum.

Anderson, J. R., & Bower, G. H. (1973). *Human associative memory.* Washington, DC: Winston.

Anderson, J. R., & Bower, G. H. (1980). Problems and new issues. In J. R. Anderson & G. H. Bower (Eds.), *Human associative memory: A brief edition.* (pp. 231-241). Hillsdale, NJ: Erlbaum.

Hinton, G. E., & Anderson, J. A. (1989). *Parallel models of associative memory* (Updated ed.). Hillsdale, NJ: Erlbaum.

Jones, G. V. (1976). A fragmentation hypothesis of memory: Cued recall of pictures and of sequential position. *Journal of Experimental Psychology: General, 105*, 277-293.

Jones, G. V. (1984). Fragment and schema models for recall. *Memory & Cognition, 12*, 250-263.

Numerical Algorithms Group. (1977). *NAG FORTRAN library manual.* Oxford and Downers Grove, IL: Numerical Algorithms Group.

Roediger, H. L. (1980). Memory metaphors in cognitive psychology. *Memory & Cognition, 8*, 231-246.

Ross, B. H., & Bower, G. H. (1981). Comparisons of models of associative recall. *Memory & Cognition, 9*, 1-16.

Rubin, D. C., & Wallace, W. T. (1989). Rhyme and reason: Analyses of dual retrieval cues. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15*, 698-709, 1989.

Rumelhart, D. E., & McClelland, J. L. (1986). *Parallel distributed processing* (Vol. 1). Cambridge, MA: MIT Press.

Schacter, D. L., Eich, J. E., & Tulving, E. (1978). Richard Semon's theory of memory. *Journal of Verbal Learning and Verbal Behavior, 17*, 721-743.

Schacter, D. & Graf, P. (1986). Preserved learning in amnesic patients: Perspectives from research on direct priming. *Journal of Clinical and Experimental Neuropsychology, 6*, 727-743.

Semon, R. (1921). *The mneme* (trans. L. Simon). London: George Allen & Unwin.

Shimamura, A. P. & Squire, L. R. (1984). Paired-associate learning and priming effects in amnesia: A neuropsychological

study. *Journal of Experimental Psychology: General, 113*, 556-570.

Shimamura, A. P. & Squire, L. R. (1989). Impaired priming of new associations in amnesia. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15*, 721-728.

PREDICTIVE UTILITY IN CASE-BASED MEMORY RETRIEVAL

Hollyn M. Johnson and Colleen M. Seifert
University of Michigan

Abstract

The problem of access to prior cases in memory is a central issue in the case-based reasoning paradigm. Previous work on thematic knowledge structures has shown that using a complete exemplar of a thematic pattern allows access to the structure and related cases in memory. However, the knowledge and expectations provided by such structures can aid in planning and problem-solving. Therefore, to be most useful, the information should become available *before* the input pattern is complete. Retrieval must therefore be possible based on only a subset of the features present in the full thematic pattern. This study investigated whether a pattern that contains elements predicting an outcome, but not the outcome itself, would result in access comparable to that found when a full pattern is used. The results showed that subjects were less successful accessing the thematic structure using partial patterns than they were when using full patterns. However, remindings based on partial patterns occurred more often than would be expected by chance. We conclude that partial patterns contain some predictive features that can allow access to a thematic knowledge structure before the pattern is complete.

Introduction

A central problem for models of case-based reasoning is the retrieval of appropriate cases from memory. Retrieval is assumed to depend on matching an input to the contents of memory, which is easier if the new case is highly similar to a case already stored, and highly distinct from other cases in memory. Two cases may be similar on different levels of abstraction; they may share surface features, such as characters and settings, or contexts, or, at the highest level of abstraction, themes. In planning and problem solving, it would be useful to access cases in memory that share abstract features with the input case because those cases could provide valuable, cross-contextual planning information. In a planning situation, optimal retrieval of past cases would be based on features that predict potential planning failures and would occur *before* the outcome of the situation has been determined. The issue, therefore, is how much information is needed to access past cases.

Recent research on reminding has focused on the role of highly abstract, thematic knowledge structures in access to individual cases (Seifert, McKoon, Abelson, and Ratcliff, 1986). The content of such abstract structures has been characterized by Lehnert (1980) as "plot units", Schank (1982) as thematic organization points (TOPs), and by Dyer (1983) as thematic abstraction units (TAUs). Such structures capture more abstract relationships between concepts, such as interactions between goals and plans, while being relatively context-independent (Schank, 1982). For example, a thematic structure about "retaliation" may include knowledge that an actor A feels wronged by an action of B and then takes action meant to displease B. The particular actors and actions are not specified; such a structure can apply both to siblings tattling on each other and gangland killings. Thus, thematic knowledge structures can be applied cross-contextually, reflecting a similarity in the point of two episodes that may have few contextual features in common (Schank, 1982).

The form and content of such structures depends on, and determines, their function. TOPs help one understand new episodes similar to old ones, provide expectations about the episode, and let one predict what is to come. TAUs, a subset of TOPs, focus more specifically on knowledge about failures in planning situations. TAUs are based on interactions of goals and plans that are reflected in adages shared by a culture, especially those involving expectation and planning failures. They then contain planning information about problems that can occur, and how to avoid or solve them. Both TOPs and TAUs organize storage of individual episodes in memory, so that cases having the same theme are stored similarly in memory.

For these structures to be useful, they need to be accessed at an appropriate time to provide useful information when processing a new, similar case. Schank (1982) proposes that understanding a new situation requires, first, categorizing the input according to a structure in memory and second, retrieving an episode that is similar to the input. In general, to make use of any expectations and predictions a knowledge structure contains, the input must be matched to a structure similar to it.

The present study looks how much of the thematic pattern need be present in the input to result in a reminding. The "whole structure" argument says that the input and the retrieved episode must share *all* abstract features in a reminding; one cannot retrieve an episode based on only *some* of the input's abstract features. If the partial and the whole structure led to retrieval of the same exact episode, this would imply that the abstract features left out of the partial structure were irrelevant or coincidental. (Birnbaum, 1986). However, according to this argument, a change in the abstract relationships within the input will affect the likelihood of retrieving a certain instance, and a reminding appropriate to one input is likely to be inappropriate when elements of the input are changed.

However, if one must receive the whole thematic pattern in an input before a TOP or TAU could be activated, this would severely limit the usefulness of these knowledge structures. The expectations, predictions, and planning information they contain could not be used until the situation had played itself out. For example, activating the TAU "closing the barn door after the horse has gone" *after* one has made this planning error would not allow one to take advantage of that knowledge and avoid the failure. To be maximally useful, a TAU should be activated *before* the outcome of a planning situation; this would require accessing the TAU based on recognizing a partial thematic structure in the input.

The present study used stories based on TAU structures to test the whole structure hypothesis, using a reminding paradigm. Subjects read TAU-based stories in a study phase, and then in a test phase were presented with stories based on the same TAUs as in the study phase. Each TAU-based test story had two versions: a partial version with information that set up conditions for a planning failure but did not allow certain identification of a theme, and a full version that completely duplicated the partial but also included the outcome of the story, thereby providing a complete thematic pattern. If the partial structure stories led to remindings at a chance rate, this would support the total structure hypothesis (and cast some doubt on the psychological validity of TAUs as planning aids). On the other hand, if partial and full theme stories result in remindings at an equal rate, one could conclude that a whole structure is unnecessary (at least in some cases). Or, finally, the results could show that having the full structure is better than just having the partial, but having the partial results in reminding at least some of the time, providing predictive functionality.

Method

Fifty-four University of Michigan undergraduates participated in a single session lasting about one hour. They received course credit in an introductory psychology class for participating. Subjects were run in groups of two to six.

The materials consisted of a study packet, a test packet, and a distractor task. The main plot of each story featured a different abstract thematic pattern reflecting a planning failure, such as "don't count your chickens before they've hatched." In addition, each story contained different contextual details, such as taking place in a school setting or involving family interactions.

Twelve different stories were used in the test packets. Four stories were used as fillers, each having themes distinct from those featured in study packet stories and from each other. Each of the remaining eight test stories had the same theme as one of the stories from the study packet. That is, the theme of the study stories, such as "counting your chickens before they've hatched," was recreated in a test story, but in a different contextual setting. Therefore, each study story and its paired test story had only the thematic pattern in common. No other similarities in character or setting were included. Each test story had two versions: a partial version, which included the initial elements of the thematic pattern but not the conclusion or resolution of the story, and a full-length version, which completely duplicated the partial version but also provided the conclusion or outcome of the story. Each partial story provided enough information to set up conditions under which the planning failure reflected in its theme could occur. However, not enough information was included to allow certain identification of a theme. An example study story, partial test story, and full-length test story are shown in Table 1.

Table 1
Sample story materials illustrating experimental manipulations.

Study Story:
Dr. Popoff depended on his graduate student Mike's hard work in the lab, but he knew that Mike was unhappy with the research facilities available in the department. Mike had requested new equipment on several occasions, but Dr. Popoff always put off Mike's requests. One day, Dr. Popoff found out that Mike had been accepted to study at a rival university. Not wanting to lose a good student, Dr. Popoff hurriedly offered Mike lots of new research equipment. But by then, Mike had already decided to transfer.

Test Story - Partial Version
Melissa's aging grandmother live in a nursing home in a nearby town. Melissa and her grandmother had always been close and dearly enjoyed seeing each other. Lately, she knew, her grandmother's health had been worsening. Her grandmother often called from the home and asked Melissa to come and visit, but Melissa wondered when she would ever find time for a visit in her busy work schedule.

Test Story - Full-length Version
Melissa's aging grandmother lived in a nursing home in a nearby town. Melissa and her grandmother had always been close and dearly enjoyed seeing each other. Lately, she knew, her grandmother's health had been worsening. Her grandmother often called from the home and asked Melissa to come and visit, but Melissa wondered when she would ever find time for a visit in her busy work schedule. When Melissa discovered that her grandmother had become gravely ill, she finally decided to go and pay her grandmother a visit. When she got to the hospital, her grandmother was in the intensive care unit and could not receive any visitors.

Each subject received four partial test stories corresponding to four of the study stories, and four full-length test stories corresponding to the remaining four study stories. The pairing of study story to full-length or partial test story was counterbalanced across subjects (i.e., subjects received different groupings of partial stories, not always the same four). Full-length and partial stories were placed in randomized order for each subject.

Each of the 12 test stories was followed by instructions, taken from Gentner and Landers (1985) and two response spaces were provided to allow for multiple remindings for each test story. The distractor task was a packet of eight insight problems to be solved, unrelated to the themes in the stories.

Subjects received a study packet and were instructed to read through the stories and study them for five minutes, as they would be tested on them later. All subjects read all the stories at least once in this study period. After five minutes the study packets were collected and subjects were told to work through problem-solving packets. They were instructed to time themselves on each problem, using a digital clock provided.

The subjects were interrupted after ten minutes of work on this task. The subjects were then given the test packets. They were verbally told there were not answers to each reminding problem in the test packet so they were not to "wrack their brains" for answers, and if one didn't come to them relatively quickly they were to go on to the next page. They were also told to work forwards through the packet and not look back at completed pages.

After finishing this phase, free recall memory for the study stories was tested. Subjects were asked to write down, on the back of the test packet, brief phrases identifying stories they could recall from the study packet. They were given a few minutes to do this and were not allowed to look back through the packet.

Results

Each page of the test packet was coded by story type (full theme, partial, or filler) and response type (match, intrusion, or null). Each subject received four of each story type in the test booklet. Mention of the corresponding story from the study packet counted as a match, whether it occurred in the first or second response space on the page. Filler stories by definition had no matches. An intrusion was coded if the response indicated a story in the study set that did not have the same theme as the test story. A null was coded if the subject made no response or made a response that could not be identified as one of the target stories (false alarms). False alarms accounted for less than 1% of the responses. The same quantitative and qualitative coding used with subject data was used for materials data as well. The filler stories had a total of 54 observations; the full theme and partial stories averaged 27 observations each.

Subjects on average wrote down 4.4 responses as remindings and could recall a mean of 4.8 study stories at the end. The overlap of these two measures was high. The number of study stories available to each subject was the number of different stories that were reported as recalls or remindings (or both); this mean was 5.5.

The mean number of responses by story and response type are shown in Table 2. Full theme stories were matched significantly

more often than partial theme stories with subjects as a random factor, $t(53) = 3.80$, $p < .0001$, but this difference only showed a trend toward significance with materials as a random factor, $t(14) = 1.76$, $p < .101$.

Table 2
Proportion of responses by story and response type.

STORY TYPE	RESPONSE TYPE	
	match	intrusion
full theme	.48	.18
partial	.34	.13
filler	------	.12

To determine whether the number of responses to full themes and to partial themes differed from chance, these means were tested against estimates of chance performance. Subjects' reported thematic and partial matches were corrected by a factor estimating chance performance, based on the probability of reporting any story from the study set (.125) and the number of responses each subject actually made in the test phase of the study. Both theme matches and partial matches were significantly different than this chance estimate, $t(54) = 10.07$ and $t(54) = 8.22$, respectively. A second estimate, looking at performance only on trials where the subject had the matching story available in memory, showed that both full themes and partial ones were matched at higher-than-chance level, $t(54) = 8.23$ and $t(54) = 7.644$, respectively.

The story types did not differ in proportion of intrusion responses when analyzed by either subjects or materials [$F(2,53) = 2.48$; $F(2,17) = 1.01$, respectively], but did differ in proportion of null responses under both analyses: for materials, $F(2,17) = 19.365$, $p < .0001$, and for subjects $F(2,53) = 113.75$, $p < .00001$. Filler stories evoked the fewest responses.

The test stories differed in length, in terms of number of words, as well as differing in type (as defined above), so analyses were done comparing long stories (full theme and filler) with short ones (partials). The number of responses, by story length, was significantly different when analyzed with subjects as a random factor, $t(53) = 2.72$, $p < .009$, with more responses being made to the shorter stories (all thematic) than to the long stories (half thematic and half not). A materials analysis of length was not significant.

Two coders rated the quality of the recall protocols, using a 5-point scale based on one used in Gentner and Landers (1985). The correlation of the coders' ratings, with agreements about null responses omitted, was .78. The ratings of the two coders were averaged to get a quality rating for each response. Number of words written was also calculated for each response; this measure was correlated with the quality measure at .71.

The average rating per response, across all story types, was 3.169. Mean quality ratings by story and response type are shown in Table 3.

Table 3
Average quality of responses by story and response type.

STORY TYPE	RESPONSE TYPE	
	match	intrusion
full theme	3.40	2.94
partial	3.17	2.64
filler	------	3.13

Between subjects t-tests of quality ratings were not significant for partial versus full theme matches, full theme matches versus filler intrusions, or partial matches versus filler intrusions. Analyzed by materials, the mean quality ratings of full theme versus partial matches showed a trend towards significance, $t(14) = 2.0$, $p < .065$. The two theme conditions did not differ in quality of intrusions when analyzed by materials, $t(14) = 1.45$, $p > .05$.

A within-subjects t-test of the quality of theme matches versus the quality of partial matches was significant, $t(38) = 2.55$, $p < .015$, with full theme responses being of higher quality. Further within-subjects t-tests of the quality of partial matches versus filler intrusions and versus partial intrusions were not possible, given the low number of subjects who had both protocols necessary for the comparisons.

Quality ratings were also analyzed by length of story. Analysis with subjects as a random factor showed a trend toward significance, $t(53) = 1.65$, $p < .105$, with long stories having responses of higher quality. This was also significant with materials as a random factor, $t(14) = 2.7$, $p < .017$.

Discussion

The results provide evidence that partial feature sets can elicit remindings based on thematic structures and can be used to access relevant cases during the planning process, rather than only after the outcome, and therefore the full pattern, is determined. Access is more likely when a whole thematic pattern is present, but partial patterns were also shown to be useful. The partial feature sets involved in the stories in this study were conditions that causally implied a certain outcome, and it may be that selection of other abstract "parts" of the pattern would not have the same effect. However, these conditions, and their relationship with the parts omitted, would be ones particularly relevant to planning situations.

The whole structure argument suggests that a partial pattern is unlikely to result in remindings because it does not completely specify a unique thematic pattern. For example, one could imagine similar set-up conditions for the adages "two heads are better than one" and "too many cooks spoil the broth." According to this argument, a change in conclusion would make the reminding inappropriate; the two adages are not analogous. However, in a planning situation getting reminded of either or both would be appropriate, as either one could provide planning information helpful in avoiding a bad outcome or making a good outcome more likely. The results show that

partial thematic patterns lead to access to instances in memory at better than chance levels.

The number of responses given was not related to the length of the test story presented. Rather, subjects were able to discriminate the thematically related stories from those not so related. This was clearly true with the full-length stories, although since no partial-length filler stories were provided it cannot be concluded with certainty that equal discrimination would occur among different types of partial stories.

The quality ratings of the protocols also show that subjects had access to the themes of the study stories. Both full-length and partial thematic stories had an average rating of over 3 (basic thematic elements present), and the observations consistently rated below 3 were mainly intrusions, suggesting that when a match was made, the test story provided better retrieval cues. The difference in quality between the responses to partial and full thematic patterns may result from the test stories having more complete reconstruction cues in the full than in the partial condition.

In summary, the results supported the hypothesis that people can use a partial set of thematic features to access related themes and cases stored in memory. The partial sets were less reliable for use in retrieval than full sets were, but they allowed some successful access. This is inconsistent with the whole structure argument, which says that reliable retrieval requires having a full thematic pattern evident in an exemplar. The results suggest that partial feature sets can serve a predictive function, allowing access to knowledge and expectations stored in thematic knowledge structures before and episode is completely determined, which in turn would help in avoiding potential planning failures. Future work will investigate whether there are particular partial sets of features that result in better access than other partial sets. For example, providing the initial conditions of a problem may lead to more reminding than when only a planning decision or outcome is provided. The functionality of the reminding constrains processing such that not all features will have equally valuable predictive ability to retrieve related information while there is still time to incorporate it into planning and problem solving.

References

Birnbaum, L. (1986). Integrated processing in planning. Doctoral Dissertation, Computer Science Department, Yale University.

Dyer, M. G. (1983). *In-depth understanding: A computer model of integrated processing for narrative comprehension.* Cambridge, MA: MIT Press.

Lehnert, W. (1980). Plot units and narrative summarization. *Cognitive Science 5:* 293-331.

Gentner, D., & Landers, R. (1985, November). Analogical reminding: A good match is hard to find. *Proceedings of the International Conference on Systems, Man, and Cybernetics.* Tucson, AZ.

Schank, R. C. (1982). *Dynamic memory: A theory of reminding and learning in computers and people.* New York: Cambridge University Press.

Seifert, C. M., & Hammond, K. J. (1990). Intelligent Encoding of Cases. *Proceedings of the AAAI Spring Symposium Series on Case-Based Reasoning,* Palo Alto, California.

Seifert, C. M., McKoon, G., Abelson, R. P., & Ratcliff, R. (1986). Memory connections between thematically similar episodes. *Journal of Experimental Psychology: Human Learning and Memory, 12* (2), 220-231.

Episodic Memory in Connectionist Networks

Chris A. Kortge
Dept. of Psychology, Stanford University

Abstract

A major criticism of backprop-based connectionist models (CMs) has been that they exhibit "catastrophic interference", when trained in a sequential fashion without repetition of groups of items; in terms of memory, such CMs seem incapable of remembering individual episodes. This paper shows that catastrophic interference is not inherent in the architecture of these CMs, and may be avoided once an adequate training rule is employed. Such a rule is introduced herein, and is used in a memory modeling network. The architecture used is a standard, nonlinear, multilayer network, thus showing that the known advantages of such powerful architectures need not be sacrificed. Simulation data are presented, showing not only that the model shows much less interference than its backprop counterpart, but also that it naturally models episodic memory tasks such as frequency discrimination.

Introduction

One of the most obvious areas of application for connectionism is in modeling memory. McClelland & Rumelhart (1986) and others (e.g. Anderson, 1972) have shown how important properties of human memory--such as content addressability and prototype extraction--fall naturally out of parallel distributed models. Furthermore, such models have the potential of ultimately linking biological and psychological accounts of memory. However, as pointed out by Grossberg (1987) and McCloskey & Cohen (1989), one of the most powerful and popular classes of CMs--multilayer networks coupled with the backpropagation learning rule, or BPCMs for short--has had difficulty accounting for a fundamental aspect of human memory: the ability to learn and remember based on a single trial.

While BPCMs can learn a single pattern to a high degree of precision, they have seemed unable to do this for a *series* of such patterns without "forgetting" all but the last few. People certainly show retroactive interference to some degree, but as they can clearly remember more than the last handful of items from a long list, they don't exhibit the kind of catastrophic interference which seems, *prima facie*, to be inherent in BPCMs. Single-trial learning is central to many memory tasks--including all those based on a single presentation of a list, which is perhaps the most common experimental procedure--and thus no existing BPCM is a serious alternative to currently popular "global" theories of memory such as SAM (Raaijmakers & Shiffrin, 1981), and Minerva 2 (Hintzman, 1986). Ratcliff (1990) argues that serious problems exist for BPCMs in modeling recognition memory, in particular.

In this paper I propose a connectionist model of memory which can learn sequentially without catastrophic interference. I first describe the model, which is much like existing BPCMs except for its unique learning rule. Next I present simulation data showing that the model has no difficulty modeling memory tasks which are clearly episodic in nature. Finally, I use these data to argue that multilayer networks are still a viable approach to devising global theories of memory and learning.

The Model

The network architecture used by the model is a two-weight-layer "encoder" network (Ackley, Hinton, & Sejnowski, 1985), which maps an input vector of N elements to an output vector, also of N elements. Input patterns are vectors with elements of either +0.5 or -0.5, which form the activation values for the input units. Activations for other units are computed using a symmetric logistic function. Thus

$$a_j = [\, 1 / (1 + \exp(-\textstyle\sum_i a_i w_{ji})) \,] - .5$$

where a is the activation value (j indexes the current layer, i the previous layer), and w_{ji} the weight on the connection from unit i to unit j. Unlike a typical network of this type, no biases are used in computing activations.

The network's task is to learn to reproduce each presented input pattern at the output, given fewer hidden units than N. In the simulations reported here, performance after learning is measured by the "match" between the input and output vectors, where "match" is defined as the dot product between input and output, divided by N, and multiplied by 4 to give a number between -1 and 1. Match values may then be compared for different classes of items, such as those items appearing versus not appearing in a studied list. The use of such an architecture in memory modeling is not new; Ratcliff (1990) uses a very similar one in his diagnosis of the interference problem. The main contribution of the present paper centers on the new algorithm used to train the network, which is now described.

Two key points need to be made in describing the learning rule used. The first point is that, in associative network learning in general, orthogonality of inputs is sufficient for eliminating interference. This means that if pattern A is trained to give an output A*, subsequently training pattern B to produce *any other output* will not change the A-to-A* mapping as

long as B is orthogonal to A. This is true in most one-weight-layer networks, and as elaborated below, can be made true to a large degree in multi-layer nets as well, by careful selection of parameters. Whether it is reasonable to assume that a model's inputs will all be orthogonal to one another, though, is another question, and is addressed in the Discussion section below.

The second key point is that gradient descent (see Rumelhart, Hinton, & Williams, 1986) is not necessary for reducing the error on a given pattern. If we look at learning as moving through the space of possible weight combinations, only one direction corresponds to doing gradient (steepest) descent. On the other hand, *half of all possible* directions will still reduce the error, although not as fast as following the gradient would. This idea is quantifiable--the dot product of the actual change in the weights with the gradient indicates the speed with which the error is reduced. The important thing here, though, is that by relaxing the constraint that the gradient be followed (that error be reduced as fast as possible), and instead only requiring that the error be reduced to *some* degree, we gain "room" for adding another constraint to our network's learning process--namely, one which reduces interference.

The standard backpropagation learning rule, which uses gradient descent, operates essentially by associating activated units in one layer with desired outputs at the next layer (to the degree the current outputs are wrong). The idea behind the present algorithm, on the other hand, is roughly to use a *subset* of the activated units in the association. In particular, only the "novel" activations are used, as defined below. Conceptually, the idea is this: when the network makes an error, we would like to blame just those active units which were "responsible" for the error--blaming any others leads to excess interference with other patterns' outputs. And, the argument goes, because "familiar" things are well learned (by definition, in a memory system) we might reduce interference with well-learned information by assuming that the novel aspects of the input are "responsible" for the output error; that is, by changing weights only from them, and not from the familiar parts.

The precise definition of "novelty" is crucial, of course. The basic approach, though, is to substitute a "novelty vector" for the actual activations during standard backprop learning (not during feedforward processing). Thus the change to a particular weight, w_{ji}, is now

$$\Delta w_{ji} = \eta \, \tilde{a}_i \, \delta_j ,$$

where $\tilde{a}_i$ is the "novelty" of the activation of unit i, η is the standard learning rate parameter, and δ_j is the delta signal as computed by standard backpropagation. This will allow both of our desired constraints to be met: First, the novelty vectors (one for the input layer, and one for the hidden layer) are chosen so that each tends to be orthogonal to highly familiar activation patterns at that layer. This property will allow reduced interference when the weight changes are made, by insuring that weights from well-learned patterns are changed less. Second, a novelty vector is generally a "part" of the actual activation vector it replaces, in some sense. As elaborated below, this insures that when we create associations by changing the weights just from the novel parts, we are still reducing the error on the pattern at hand (although not as fast as with gradient descent).

The novelty vectors used were as follows: The novelty vector for the input layer is simply the input (target) vector minus the output vector. This makes intuitive sense: because the task of the network is to reproduce previously seen inputs at the output, the output error should provide a rough indication of which aspects of the input are novel. As for the hidden layer, its novelty vector is obtained by feeding the input novelty vector through the first layer of connections (just as if it were another input pattern), and using the resulting hidden unit activations as the hidden novelty vector. (As explained below, the hidden novelty vector is taken to be zero in certain cases.) The justification for using this particular vector is not strong; it is simply that this hidden novelty should tend to correspond to hidden activation patterns which have not previously occurred (because the input novelty which produced it is *itself* a pattern which isn't highly learned).

These particular definitions of "novelty" are not necessarily optimal; they are only heuristics. They do possess the characteristics described above, though. First, they tend to be orthogonal to highly familiar activation patterns at the corresponding layer, even when the current activation pattern is *not* orthogonal to previous ones. This is because (1) output errors tend to be orthogonal to well-learned outputs, and (2) since an encoder network is being used, and thus there is a correspondence between inputs and outputs, we can readily define input novelty in terms of output error.

Second, these novelty vectors satisfy the constraint that the new algorithm still reduce the output error. It can be shown that substituting *any* vectors for the activation vectors during backpropagation learning will still allow the error (which is computed using the *true* activations) to be reduced, *if* (but not only if) each activation vector **a** is replaced by a vector *correlated* with **a**. It is readily observed that when using binary +.5/-.5 targets, as here, the sign of *target* − *output*, the input novelty, is always the same as the sign of *target*, for each unit. Thus the input activations and the input novelties are correlated. As for the hidden novelty vector, it also tends to be correlated with the actual hidden activations.

However, this is not guaranteed, so the hidden novelties are set to zeros when this is not true (an infrequent occurrence), thus restricting learning to the first weight layer.

In sum, then, this new learning algorithm is much like BP, but with a more focused assignment of blame. By blaming novel things more than familiar things when an error occurs--where "novelty" is determined by the evolving network memory itself, in a context-dependent manner--the network can modify to a greater degree those connections which don't much affect the storage of well-learned patterns. As a result, storage of a new pattern requires relatively little disruption of existing knowledge. Because the gradient is not followed, this advantage is obtained at a cost of lower learning speed. However, "learning speed" in this case means *within a pattern presentation*, which in human terms might correspond to the amount of time a stimulus is viewed by a subject. But since we have no idea how much network learning (e.g. magnitude of weight change) should correspond to a given stimulus duration in a human learning experiment anyway, this is not a problem for the model. What is important is that the present model exhibits improved learning *across* patterns, in that old patterns need not be represented many times to allow remembering of a long list. This is shown next.

Simulation Data

Forgetting functions

Perhaps the simplest way to assess interference is to present a list of patterns, training each pattern in turn by some amount, and, after one run through the list, to examine performance as a function of serial position of the list items. This is the basic procedure used in many experiments on human memory, and is also the procedure Ratcliff (1990) argues must be used in modeling such experiments (i.e., if a list is not repeated for a person, it should not be repeated for our network model). The present model was tested against standard BP on such a task, using identical network architectures. These networks were as described above, using 32 input, 16 hidden, and 32 output units (a "32-16-32" net). For each simulated subject, 16 binary patterns were constructed--random vectors, with elements +.5 with probability 1/2, otherwise -.5.

For each run, 15 of the 16 patterns were actually learned. The learning rate, η, was .5, and the initial weights of the network were normally distributed with mean zero and variance (σ^2) = .25. Each pattern in turn was repeatedly presented--meaning simply that multiple learning steps were taken on each--until either the mean squared error over the output units reached .01, or a pre-set maximum of 1000 steps was reached. (From now on, each set of steps on a single pattern will be considered one "presentation".) After the 15 pattern presentations, the match value between input and output was tested for each pattern, including the unpresented one. There were 250 simulated Ss in each condition. Figure 1 shows the resulting serial position curves for the novelty rule, backprop, and another instantiation of backprop as tested by Ratcliff (1990), which only studied eight single items.

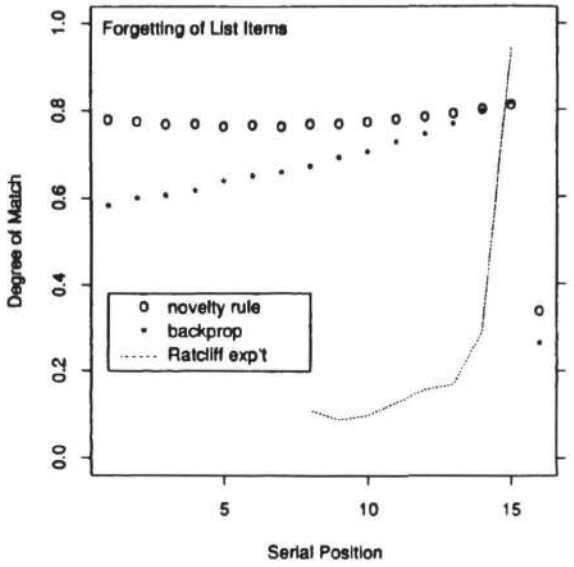


Figure 1. Degree of input/output match as a function of serial position, after one list presentation. Position 16 represents the unpresented item. The Ratcliff data begins at position 8 for comparison purposes.

The first thing to note in the figure is that there is much more forgetting in the Ratcliff experiment than in either of the others. Two aspects of Ratcliff's network probably contributed to its worse performance: First, asymmetric (0 to 1) activation functions were used, which induces a larger average correlation between activation vectors. This generally leads to higher interference, as already noted. Second, smaller starting weights were used (uniform +.3 to -.3). David Rumelhart (personal communication) has suggested that larger initial weights tend to lead to lower interference, because training tends to move activations further into the tails of the sigmoid function. Because the derivative of the sigmoid is near zero there, subsequent changes of weights to the "saturated" units will be small, meaning that interference with the patterns represented by those units will be less. This analysis was supported by comparing separate runs of backprop on the present experiment using σ^2 values of .01, .25, and 1.0.

It is also apparent from the figure that forgetting is even less when using the novelty-based learning rule.[1] This result must be qualified, though. Further

[1] In addition, there appears to be a slight primacy effect with the

simulations showed that both backprop and the novelty rule improved significantly on this task with increasing network size. It may well be, then, that with a large enough net there would be virtually no interference with *either* algorithm on this task. Furthermore, importantly, the question of what size net is "proper" for modeling memory remains open, as noted in the Discussion section below.

However, while backprop improved greatly with larger nets, the patterns used were also uncorrelated on average, which may have helped. The next experiments used patterns which were distortions of a single (random) prototype pattern, and thus were explicitly correlated. Each distortion was generated by flipping any given bit of the prototype with probability .25. Figure 2 compares the serial position functions obtained with such patterns for two sizes of backprop networks (32-16-32 and 128-64-128), and the 32-16-32 novelty network used above. Other parameters were as above.

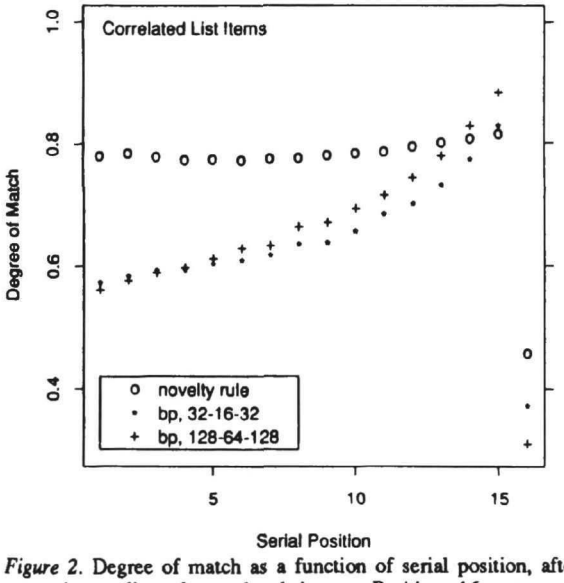


Figure 2. Degree of match as a function of serial position, after presenting a list of correlated items. Position 16 represents untrained items which were correlated with the presented items.

Comparing the two backprop nets, while the large-network curve ends up higher (possibly due to a tendency to produce more learning per step), its slope is also steeper; thus overall, forgetting is more pronounced if anything for the larger net (note that new-item match is lower for the larger net, though). A 256-128-256 net, and nets with larger σ^2, gave similar results. These results were perhaps predictable: While forcing activations further into the sigmoid tails can reduce interference--by making delta vectors more orthogonal to learned outputs--this should not reduce any interference caused by correlated *inputs*. There is no apparent way to remove correlations among the inputs simply by changing the network configuration.

The novelty-rule network, on the other hand, exhibited very little interference even with these correlated items. Moreover, further runs showed the novelty rule to *improve* with network size: absolute match values were approximately unchanged, but new-item match values decreased significantly, so that forgetting in terms of discrimination was less. Thus it appears that even with correlated patterns, multilayer network models need not suffer from extreme interference.

Varying amount of learning

In the next experiment, discrimination between "old" (studied) and "new" (unstudied) items was examined as a function of amount of learning, as measured by number of learning steps taken, or the "duration" of each pattern presentation. This is important to check, because while human data shows increasing discrimination with increasing item learning time, such an improvement is not predicted by some current memory models. Simple linear models, for example, predict that the variance in output to new items increases with the mean old-item output as learning amount increases, so as to keep d' constant (Shiffrin, Ratcliff, & Clark, 1990). Also, Ratcliff (1990) has shown that several variants of the backprop-based encoder network do not predict a strictly increasing d' (the actual patterns are complex, often nonmonotonic, and dependent on the parameters used).

A 32-16-32 architecture was used as in the forgetting simulations. Other parameters were $\sigma^2 = .25$, and $\eta = .1$. However, rather than training each list item to a criterion as before, each item was simply given a fixed number of learning steps before proceeding to the next item. This number of steps was varied (between lists) from 1 to 512, by powers of 2. Two types of patterns were tested, for both standard backprop and the novelty variant: "uncorrelated", with elements +.5 and -.5 equally likely; and "correlated", where list items were distortions of a single prototype, as described above. Within each learning amount, 100 lists of 16 items were trained for each condition. Performance was compared on the learned items and 16 new items using d' (mean old-item match minus mean new-item match, divided by standard deviation of new-item match), where new items were generated the same way as list items (from the same prototype, in the "correlated" conditions).

As depicted in Figure 3, d's were increasing for the most part. (The very slight decrease over the last interval in the novelty/correlated condition was insignificant.) Furthermore, in a replication of all

novelty rule, with items from the beginning of the list giving better matches than those in the middle. More work is necessary to determine the significance of this.

conditions, using $\sigma^2 = 1.0$, there were *no* decreases, and a similar overall pattern was obtained. Note that in all conditions the increases extended into or past the range of d's typical in human experiments (about 1 to 3). Mean match values after training the 512-step lists ranged from about .70 to .78. Backprop was somewhat worse overall than the novelty rule, presumably due to interference effects. In sum, Ratcliff's conclusion that increasing d' is a problem for the encoder network must be qualified. In particular, further simulations suggested that initial weight variance is critical: using $\sigma^2 = .01$ rather than .25, nonmonotonic functions much like Ratcliff's were obtained with backprop (the novelty-rule function was still monotonic). There is a degree of this nonmonotonicity apparent in the bp/uncorrelated condition in the graph, as well.

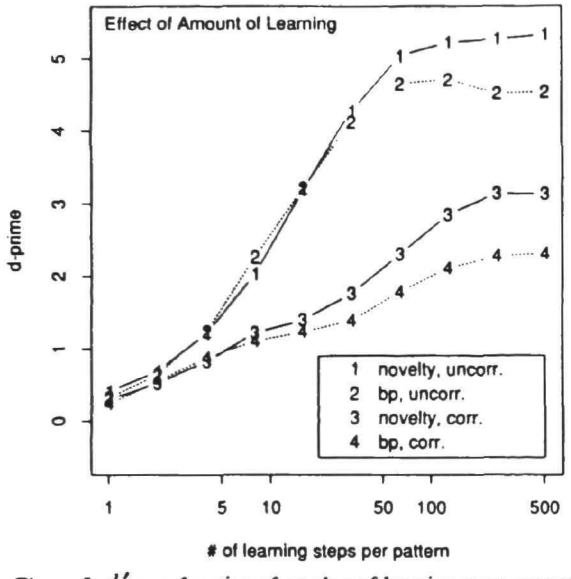


Figure 3. d' as a function of number of learning steps per pattern in a once-presented list. Note the logarithmic scale.

Frequency discrimination

After being presented with a list of items, some of which are repeated various numbers of times (at various intervals), people can give fairly good estimates of the frequencies with which items were presented. This task can be viewed as a generalization of the recognition task, wherein the possible frequencies are limited to zero and one. Hintzman (1988) has called frequency judgment "a quintessentially episodic memory task", presumably because information concerning individual item presentations (namely their occurrence) is required in order to respond correctly.

The present novelty-based model was applied to this task, using a paradigm similar to one used by Hintzman (1988) in testing his multiple-trace memory model, Minerva 2, on frequency discrimination. A size 48-9-48 network was used, with $\eta = .5$ an $\sigma^2 = .25$. Twenty-four random patterns were generated, with four assigned to each of six frequency conditions, 0 to 5. Each pattern was copied the proper number of times, and the resulting list of 60 items was randomly permuted. Items were then trained one-by-one, with only a single learning step taken on each (thus considerably less training was done per pattern than in the forgetting experiments above). After training, match values were tested for all 24 patterns. This procedure was replicated for 5000 simulated Ss, resulting in distributions of match values for each of the six frequencies. From these distributions forced-choice data were computed, giving the proportion of errors the network would make if required to pick the more (or less) frequent item from a pair taken from the original 24. These error data are shown in Figure 4.

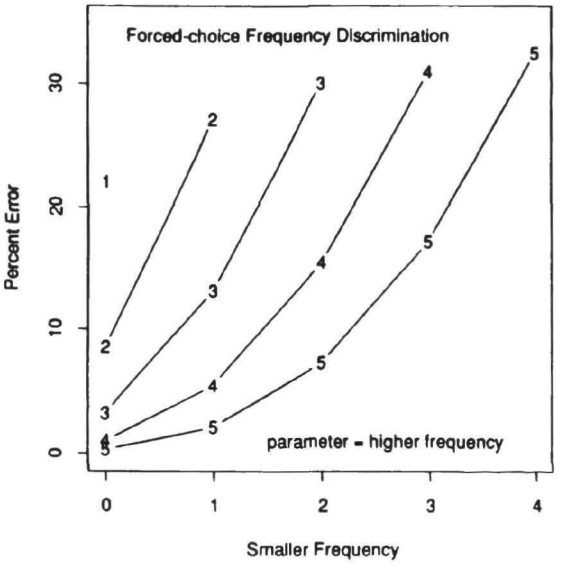


Figure 4. Errors in simulated forced-choice data, as a function of the frequency of the less frequent item in a test pair. Data were generated from distributions of match values corresponding to each frequency.

Overall the model predicts that (given appropriate response criteria) discrimination between two frequencies improves with greater frequency difference, and holding frequency difference constant, discrimination worsens with increasing frequencies. This same pattern holds with people (Hintzman & Gold, 1983), when results for the "choose the more frequent" and "choose the less frequent" instructions are collapsed. Interestingly, the results obtained here are very nearly indistinguishable from those obtained with Minerva 2; in fact, although each matches the human data well, the models match each other much better than either matches the human data. For this task, then, there seems to be no theoretical advantage to maintaining a distinct memory trace for each individual "event" to be remembered. Clearly, the present connectionist model does as well by "strengthening" each repeated item (by increasing the relevant weights). (Backprop was also able to model this particular task.)

List-specific frequency judgment

One might rightfully argue that there is more to making frequency judgments than simply assessing the familiarity of an item, though. In particular, human subjects are faced with the more difficult task of not only having to discriminate different frequencies that occurred *within* an experimental list, but *also* having to discriminate the list presentations from all the times the same items have been observed *outside* the experiment. Thus, for example, people are quite good at discriminating rare words which appeared in a list from common words which did not appear, even though presumably the "baseline" familiarity is much higher for the common words. Another example is provided by an experiment by Hintzman & Block (1971), in which people were asked to make *list-specific* frequency judgments. Even though two different lists were made up of the same items, subjects could give good estimates of an item's frequency in one list almost independently of its frequency in the other list. In a sense, this task is even more "quintessentially episodic" than standard frequency judgment; not only must information about number of occurrences be retained, but this information must be distinguishable on the basis of instructions to limit the relevant context.

The present model was tested on this task using a 120-30-120 net, with $\eta = .3$ and $\sigma^2 = .25$. An approach much like Hintzman's (1988) was used. In order to model different lists, 80 input units represented list context, and each list was assigned its own random context vector which was presented along with every item in that list. Twenty-seven random 40-element vectors were generated, and three of these assigned to each of the 9 possible combinations of the frequencies 0, 2, and 5 across the two lists. Thus each of the three "2-0" patterns appeared twice in the first list and zero times in the second list, and similarly for 0-0, 0-2, 0-5, 2-5, and so on. Once the proper number of copies were made, each list was randomly permuted and then trained, with one learning step per item.

Figure 5 shows the match values obtained after training both lists, averaged over 500 simulated Ss, along with the data from H & B linearly transformed to match the model's output values (using a least squares fit over the 9 means). Each pattern was tested separately with each context vector, and data for the two lists was collapsed.[2] Thus for example the "target-frequency = 5, nontarget-frequency = 2" data point represents match values obtained by testing the 5-2 patterns in context 1, and the 2-5 patterns in context 2.

[2] The model showed a reliable primacy effect, with higher matches overall on the first list, but it was not large and did not affect the conclusions.

In the graph, perfectly flat (separated) lines would mean that discrimination was perfect; i.e., that there was no interference from the nontarget list in making frequency judgments. Hintzman (1988) proposes measuring performance on this task by a "discrimination index" (DI), obtained by dividing the variance among means accounted for by nontarget frequency by that accounted for by target frequency. Such a measure could vary from 0 (perfect list discrimination) to 1 (no discrimination). H & B's subjects showed a DI of .097, and the present simulation gave a DI of .093. Correlation of the model with the H & B means was .998. (A total of about five simulations of this experiment preceded this particular one, so not a lot of fitting was involved.)

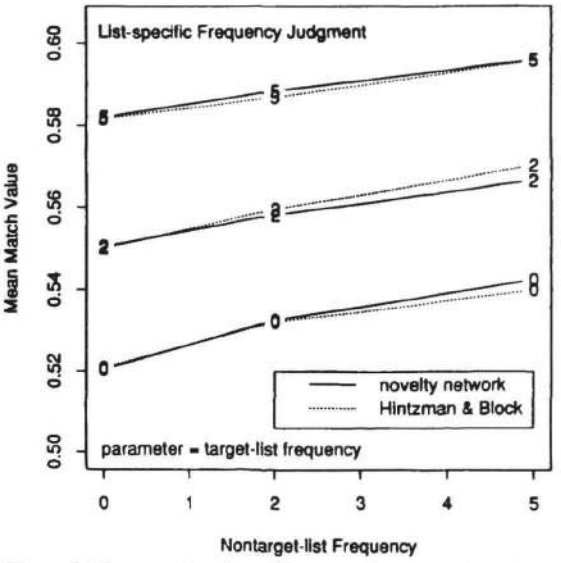


Figure 5. Mean match values ("frequency judgments") for items in a particular list, as a function of frequency of the same item in the other list. Hintzman & Block's subjects' data was linearly scaled to match the network's output values.

The present model had no trouble, then, even with this highly episodic memory task. (Backprop also had no trouble, except when interference effects were large due to use of correlated items.) The novelty-based network did require many more context elements than did Minerva 2, however, in order to match human performance (as with Minerva 2, discrimination improved with additional context elements). This might seem disadvantageous, but it isn't clear how much of a person's internal representation of an episode is devoted to "context", anyway. Moreover, Minerva 2 does not attempt to model the existence of extra-experimental memories. It seems reasonable that the more occurrences of an item are stored in memory, the larger must be the number of possible contexts in order that any small subset of those memories be accessed. Put differently, the more background noise, the harder the discrimination task becomes. Thus a version of Minerva 2 which modeled prefamiliarization might require more

context elements as well. The present model, on the other hand, does all its learning in terms of what is already known; i.e., all previous knowledge is contained in the model, although here this pre-existing memory was modeled simply with random weights. Thus it has more of a burden in trying to discriminate the experimental presentations from background "memories".

Discussion

As a whole, the results presented here argue strongly that episodic memory tasks are not beyond the modeling capabilities of multilayer networks. First, it was shown that the catastrophic interference exhibited by many of these networks is readily avoided. Three different tricks were used to accomplish this: (1) increasing the variance of the starting weights, which, by making more use of the nonlinearity of the activation functions, tends to protect from change those weights leading to units being "used" to represent old items; (2) increasing network size, which provides more units to *be* "used" in representing old items, and also increases the variance of initial activations (by increasing the fan-in to the units) which should have an effect similar to increasing the initial weights; and (3) using a new learning rule, which increases orthogonality for learning purposes while still reducing error. While use of the first two tricks suggested that arbitrarily low interference might be obtained with backprop using uncorrelated (on average) patterns, only by using the new learning rule did this seem possible for explicitly correlated patterns.

Second, it was shown that the increasing d' exhibited by people with increased learning per item can be modeled by backprop, within certain parameters, and by the novelty-based learning rule, with no observed need for limiting parameters. This attenuates the conclusion made by Ratcliff (1990) that this is a problem for backprop networks. (Ratcliff does note the qualitative differences changes in parameters can make.)

Third, the novelty-based network (and backprop to some extent) was shown capable of modeling a simple frequency discrimination task, and a more difficult list-specific frequency judgment task. While, unlike recall tasks, these tasks are based only on scalar "familiarity value" outputs, they still depend on the ability to maintain information about individual episodes, and thus seem like good indicators of episodic memory.

Disadvantages

Because of the preliminary nature of the present model, it is hard to say much about its explanatory disadvantages. It is reasonable to ask at least, though, whether the tricks used herein are justified from a theoretical standpoint. Increasing initial weight variance doesn't seem problematic: people clearly bring a lot of knowledge to an experiment (whether random weights capture this knowledge is debatable, of course). Increasing network size seems fairly reasonable as well; obviously the number of a person's neurons dwarfs the number of items in any memory experiment. However, it must be noted that, performance level held equal, smaller networks seem to generalize better to new experience than large ones (consistent with general overfitting arguments). Note, though, that "interference" and "generalization" are in essence two sides of the same coin. Each implies learning of one pattern affecting the output of another; the difference is whether such effects are "good" or "bad", given the task at hand. It is an open question, then, to what extent future CMs will exhibit high "good generalization" and low "bad generalization".

As for the novelty-based learning rule, no obvious explanatory disadvantages (relative to other CMs) have surfaced yet. Indeed, it seems clear that people *do* tend to focus, in some sense, on unusual aspects of their environment. Furthermore, it seems like successive inputs to a human memory system would tend to be highly correlated, if anything, based on continuity of the environment. Thus the novelty rule (or something similar) may well be necessary for modeling human memory in multilayer nets without high interference. On the negative side, preliminary runs suggest that the novelty rule is not as good as BP at storing information in small (relative to the task) networks, when very large amounts of learning per pattern are used. This need not be problematic for modeling, though, since as noted there is no pressing theoretical limitation on a modeling network's size.

Relation to other models

Several points need to be made regarding previous models. First, a large number of connectionist (or connectionist-flavored) models do not suffer from the catastrophic interference problem as do BPCMs. These include holographic models such as CHARM (e.g. Eich, 1982), and the various ART models (e.g. Grossberg, 1987), among others. The present research speaks to these models only indirectly, insofar as it lends support to a class of potential competitors (multilayer error-reduction networks). It is also important to note that the basic idea behind the learning rule introduced here--that focusing on novelty in learning can reduce interference--is not new. Grossberg (1987) has argued that this is necessary to allow stability of learning in a changing environment. Otwell (1990) has used substitution of novelty vectors (of a different sort than here) in backprop learning. And in an intriguing parallel, Holyoak, Koh, & Nisbett (1989) use an "unusualness heuristic" in generating "exception rules" in their rule-based theory of animal conditioning, which

function to "censor useful but imperfect default rules, protecting them from loss of strength". This seems roughly like another way to say "reduce interference". The present paper, then, might be used to argue that the difference between rule-based and connectionist accounts of learning is not so clear as Holyoak et al. suggest.

There are many similarities between the present model and Hintzman's multiple-trace model, Minerva 2 (1988). For instance, inputs are represented as feature vectors in each, and each allows for two kinds of outputs: a scalar familiarity value, and a vector representing a "retrieved memory". These similarities have led to the frequency judgment experiments herein being modeled much as in Minerva 2. The models' storage assumptions, however, are quite different: while Minerva 2 maintains distinct memory vectors corresponding to each experienced "event", the present model, as other CMs, superimposes all memories on a single set of connection weights. Thus it is striking that the agreement in their predictions noted above is as strong as it is.

Hintzman allows for the possibility that associative matrix models might account for the range of frequency-judgment data exhibited by people, but opts for a multiple-trace view for various reasons--such as the accessibility of individual event information, and the ability to activate individual traces as a nonlinear function of their similarity to a probe (Hintzman, 1988; 1986). The present research argues, though, that given some distinguishing context cues, individual event memories can also be accessed by a network. Also, because of the nonlinear activation functions of the present model, it too has the potential to exhibit nonlinear generalization as people do (pilot work on prototype extraction experiments has supported this claim). Thus there is no *obvious* reason why multiple-trace theories such as Minerva 2 should possess any inherent explanatory advantage over CMs.

Conclusion

It has been argued herein that a potentially major problem with multilayer network models--that they exhibit catastrophic interference--is in fact not inherent in these models at all. By using the learning rule introduced here, and in certain cases using even simpler devices, these models can learn individual associations to a large degree with little disruption of prior knowledge. Given this, there is no obvious reason why an encoder network-based alternative to current global models of memory could not evolve. Of course, the present paper has barely scratched the surface of such an undertaking, and there is much more to be said, surely both pro and con, about such memory models. Nonetheless, a non-obvious capability of these networks--the ability to learn, remember, and later make use of information about individual event information--has been shown to exist. Taken together with previous connectionist work, and the tantalizing prospect of meshing biological and psychological accounts of memory, this suggests strongly that connectionism has a solid place in future memory research.

Acknowledgments

I wish to thank David Rumelhart for his ongoing help, as well as helpful input on the present paper, and David Bryant and Gary Cottrell for their extremely helpful reviews. I also thank Roger Ratcliff for providing me with a draft of his paper, and apologize in advance for any confusion arising from my use of this draft rather than the published version.

This research was partly supported by a National Science Foundation Graduate Fellowship. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of the National Science Foundation.

References

Ackley, D. H., Hinton, G. E., & Sejnowski, T. J. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science, 9*, 147-169.

Anderson, J. A. (1972). A simple neural network generating an interactive memory. *Mathematical Biosciences, 14*, 197-220.

Eich, J. M. (1982). A composite holographic associative recall model. *Psychological Review, 89*, 627-661.

Grossberg, S. (1987). Competitive learning: From interactive activation to adaptive resonance. *Cognitive Science, 11*, 23-63.

Hintzman, D. L. (1986). "Schema abstraction" in a multiple-trace memory model. *Psychological Review, 93*, 411-428.

Hintzman, D. L. (1988). Judgments of frequency and recognition memory in a multiple-trace memory model. *Psychological Review, 95*, 528-551.

Hintzman, D. L. & Block, R. A. (1971). Repetition and memory: Evidence for a multiple-trace hypothesis. *Journal of Experimental Psychology, 88*, 297-306.

Hintzman, D. L. & Gold, E. (1983). A congruity effect in the discrimination of presentation frequencies: Some data and a model. *Bulletin of the Psychonomic Society, 21*, 11-14.

Holyoak, K. J., Koh, K., & Nisbett, R. E. (1989). A theory of conditioning: Inductive learning within rule-based default hierarchies. *Psychological Review, 96*, 315-340.

McClelland, J. L. & Rumelhart, D. E. (1985). Distributed memory and the representation of general and specific information. *Journal of Experimental Psychology: General, 114*, 159-188.

Otwell, K. (1990). Accelerating backpropagation learning with novelty-based orthogonalization. *Proceedings of the International Joint Conference on Neural Networks*, Jan. 1990, Vol. 1. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Raaijmakers, J. G. W. & Shiffrin, R. M. (1981). Search of associative memory. *Psychological Review, 88*, 93-134.

Ratcliff, R. (1990). Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions (DRAFT). To appear in *Psychological Review, 97*, March 1990.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by error propagation. In Rumelhart, D. E., McClelland, J. L., & the PDP Research Group, *Parallel Distributed Processing*, vol. 1. Cambridge: MIT Press.

Shiffrin, R. M., Ratcliff, R., & Clark, S. E. (1990). List-strength effect: II. Theoretical mechanisms. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 16*, 179-195.

On the Domain Specificity of Expertise in an Ill-Structured Domain

Colleen M. Zeitz

Learning Research and Development Center
University of Pittsburgh

An important issue concerns the relation between expertise in highly-structured domains and ill-structured domains. This study explored the information processing abilities associated with expertise in literature, an ill-structured domain. Literary experts were superior to novices in gist level recall, the extraction of interpretations and the breadth of aspects addressed of literary texts but not of a scientific text. The results indicate that expertise in literature appears to share features with expertise in highly-structured domains, including domain-specificity and an abstract level of representation.

Our understanding of expertise is based primarily on studies of highly-structured domains. However, research is suggesting that the nature of expertise in ill-structured domains may be different (Glaser, in press; Voss, Green, Post & Penner, 1983). One way to examine this issue is to intensively study experts in a less-structured domain, discovering what characteristics they share with experts in other domains and where they diverge. Literature is a good candidate for such an endeavor, because the domain is a rich, complex one that is neither quantitative nor rule-based. Domains that have been studied by cognitive scientists tend to have mathematical rigor, well-defined rules and clear methods for evaluating the correctness of solutions and distinguishing winners from losers (e.g., chess, Chase & Simon, 1973; physics, Larkin, McDermott, D. Simon, & H. Simon, 1980; computer science, Adelson, 1981). These domains address well-structured problems. In contrast, the domain of literature can be characterized as ill-structured. This is not to say that the field itself is chaotic, but that its practitioners apply themselves to open-ended questions, for which agreed-upon solution evaluation procedures do not exist. While rules or laws that describe the processes a literature expert uses in analyzing a text may exist, they are not transparent. One might assume that findings from well-structured domains could be applied to domains like literature. However, generalizations of this type may be premature. Glaser (1986) has suggested that our understanding of expertise may be biased, because less-structured types of domains have been largely ignored.

In contrast to the literary theorists who describe the importance of literature-specific knowledge and information-processing abilities for processing literature (e.g., Purves, 1984; Mandel, 1976), there are literary theorists who espouse more global, all-encompassing views. Some claim that all of one's experience with the world contributes to the processing of a given literary text (Purves, 1971; Fillion, 1981). Some literary theorists also suggest that the same strategies employed in real life situations are applicable to literary experiences (Fillion, 1981). Others go so far as to claim that the skills gained from the study of literature are sufficiently decontextualized to be applied to all domains (Landow, 1989). This raises the issue of whether expertise in an ill-structured domain is domain-specific, as is expertise in highly-structured domains (e.g., Hatano & Osawa, 1983).

In this study, literary experts and novices were asked to read, perform recall tasks on and analyze both literary and non-literary texts. The results are used to address whether expertise constructs from highly structured domains apply to a less structured domain like literature. Specifically, the analyses consider whether literary experts demonstrate superior memory, abstraction and reasoning with regard to literary texts, but not with regard to non-literary texts. There were three groups of subjects: 24 students in their third year of high school, 13 graduate students who were in their third or higher year in an engineering Ph.D. program, and 16 graduate students who were in their third or higher year in an English Ph.D. program.

Results and Discussion

The most widespread finding in the field of expertise research is experts' superior memory for domain-relevant material. Verbatim, gist and word level free recall of a poem and a scientific text were compared for the three groups. At issue here is whether literature experts have superior recall of specific types of texts, or texts in general, and at what levels of detail any superiority is displayed.

Scoring of free recalls. As the first stage in coding the free recalls, the poem and scientific text were divided into propositional arguments. Each item recalled by subjects was matched to propositions from the appropriate text. A match was categorized as verbatim if it contained the exact wording as a proposition from the text and it was used in an appropriate context. A match was categorized as gist if the wording differed, but the meaning had been retained. Lastly, a match was categorized as being at the word level if some of the content words from a proposition were simply listed, or used in conveying an idea not contained in the text.

Verbatim recall. Subjects' verbatim recall scores for the poem and scientific text were examined first. A 3(group) x 2(text) ANOVA revealed a significant effect for group, $F(2,50) = 9.08$, $p < .0005$. As Figure 1 illustrates, both graduate student groups recalled more at the verbatim level than the high school students. The verbatim scores for the scientific text were significantly better than those for the poem, $F(1,50) = 4.33$, $p < .05$. The interaction was not significant, $F(2,50) = 1.87$, $p > .10$.

It is not possible to conclude that the effects of group in this analysis are due exclusively to the differences in the subjects' experience in the domain of literature. One confounding factor is age. A number of literary theorists claim that age affects literature processing ability (e.g., Purves, 1971, 1984; Fillion, 1981). Another difference may be in intelligence: Perhaps not all the high school students would meet the requirements for entering a doctoral program at Brown University. Thus, the high school students differ from the other groups on a number of important dimensions. The engineering group serves as a control for these factors. Their age, years of education, and ability to meet the University's admission standards were comparable to the English students'. However, their experience with the domain of literature was comparable to that of the high school students. Thus, they provide an important control group whose comparison to the English students can lead more directly to conclusions regarding the effects of experience in the domain of literature. For this reason, each analysis was repeated using only the data from the graduate students. As the purpose of these analyses is to clarify group effects, effects that do not interact with group will not be reported. The analysis of verbatim recall was repeated without the high school students' data. Neither the main effect of group nor the interaction was significant.

Gist recall. The next analysis concerned subjects' recall of both the poem and the scientific prose at the gist level. The effects of group, text and their interaction were significant, $F(2,50) = 7.18$, $p < .005$, $F(1,50) = 5.49$, $p <$

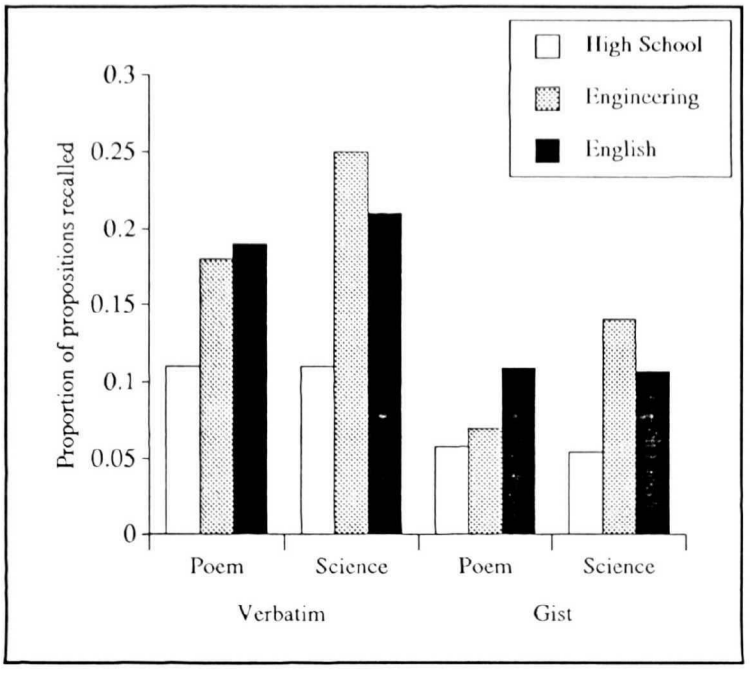


Figure 1. Proportion of items recalled from the poen and scientific text as a function of level of precision of recall.

.05 and F(2,50) = 6.45, p < .005, respectively. In Figure 1 it can be seen that the overall amount recalled by high school students appears to be less than that for either of the graduate student groups. The English students have better gist-level recall of the poem than the other two groups, whereas the engineers recalled the scientific text best.

The same analysis was repeated without the data from the high school students. The effect of group was not significant, $F(1,27) < 1$. At the gist level, the scientific text was recalled better by the engineers, and the poem was recalled better by the English students, as can be seen in Figure 1, $F(1,27) = 6.45$, $p < .05$. Thus, although the two groups of graduate students were equivalent in their overall recall, each group recalled the text within their own area of expertise better. This indicates that neither group had better overall memory for verbal material, but instead that the specialized knowledge they accumulated through experience allowed them to be more efficient at representing information pertaining to their field.

Word recall. An ANOVA for the number of items recalled at the word level did not reveal any significant effects. This category of recall was used very infrequently for all three groups and both texts (less than five percent for each).

The free recall findings suggest that the English students have superior memory for a literary text but not for a scientific text. The fact that the English students' memory was not superior for the scientific text indicates that their expertise applies to a limited set of texts, those that have the characteristics of literature, and not to all written material. Thus, these findings demonstrate the English students' domain-specific skills.

Abstraction

Experts and novices in the domain of literature may differ in the degree of abstraction in their analyses of literary texts. In order to examine these differences, during the second session subjects were asked to list the broadest range of essay ideas that they could generate. I analyzed both the breadth of the topics covered (i.e., the content of the sentences) and the depth to which they were addressed (i.e., the level of abstraction of the sentences). If the English students have deeper, more principle-governed representations of the literary texts, then these should be made manifest in the topics which they can generate regarding the literary texts, a poem and a short story. Their skills with regard to a scientific text should be attenuated because their representations of this text are no more abstract than those of the other groups'.

Scoring of topic sentence data. Subjects were instructed to make a list of the main topics they would include in an analysis of each of the texts, addressing aspects of the structure and content that they thought were important. The instructions directed them to express each of their ideas as the topic sentence of an essay on this theme and to list as many as possible. Fillion's (1981) system for categorizing students' inquiries served as the basis for the coding system. First, the coders classified the content of a given statement. The possible categories were: *events*, for events or plot; *characters*, for characters or relationships; *setting*, for setting, mood or atmosphere; *images*; *themes*, for themes or ideas; or *language*, for language, style or structure. Secondly, the coders classified the level of each sentence. Sentences were categorized as factual if they restated ideas explicitly mentioned in the story. Sentences were classified as interpretive if they offered a conclusion that could be disputed concerning some aspect of the story. The coders developed a number of additional categories for sentences which did not fit into this matrix (e.g., facts that were external to the story). Two coders independently scored all of the protocols. The coders were blind to the level of expertise of the subjects who had produced the topic sentences. All differences of opinion between the two coders were resolved through discussion.

The level of the topic sentences. The number of factual, interpretive and other statements subjects wrote were compared. The data are illustrated in Figure 2. A 3(group) x 3(text) x 3(fact, interpretation or other) ANOVA was performed. There was a significant effect of group, with English graduate students writing the most sentences, and engineers writing the fewest, $F(2,43) = 6.98$, $p < .005$. Subjects wrote the largest number of interpretive sentences and the smallest number of other types of sentences, $F(2,86) = 29.14$, $p < .0001$, but the relative production of sentence types interacted significantly with group, $F(2,43) = 15.63$, $p < .0001$. The tendency to write more interpretive sentences than factual was most exaggerated in the English graduate students and was reversed for the high school students. All of the groups wrote approximately equivalent numbers of factual and other statements for all of the texts. All of the groups wrote the most about the short story and the least about the scientific text, $F(2,86) = 23.08$, $p < .0001$. The amount written for each of the texts interacted marginally with group, $F(4,86) = 2.06$, $p < .10$. The production of different types of sentences did interact with text, $F(4,172) = 22.17$, $p < .0001$. There were more interpretive sentences than factual ones for the poem and short story, but more factual than interpretive ones for the scientific text. The three way interaction did not reach significance.

When this analysis was repeated, omitting the data from the high school students, the same pattern of significance for the results occurred, with one exception. The three-way interaction was marginally significant, $F(2,50) = 2.95$, $p < .10$. This may be due to the extremely large number of interpretive statements the English graduate students made in reference to the short story and poem. As Figure 2 shows, the English students' enormous advantage in producing interpretive statements was attenuated for the scientific text. This is an indication that their

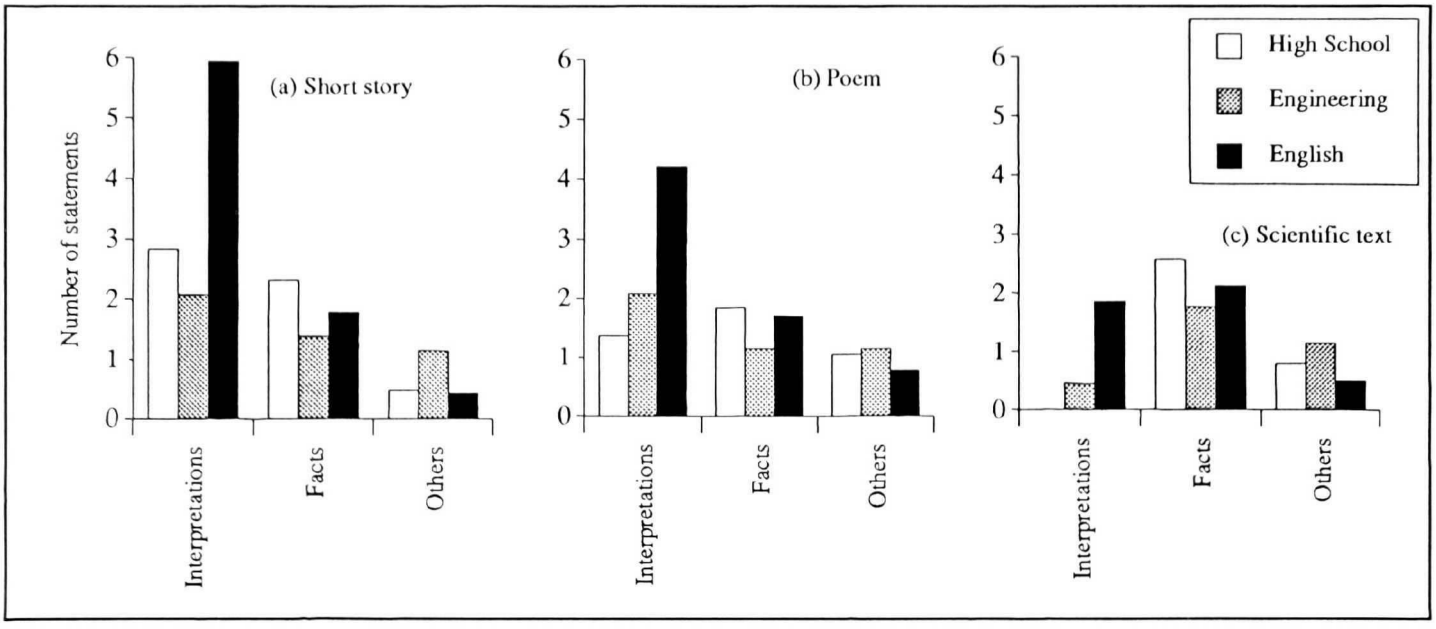


Figure 2. Number of topic sentences as a function of level of abstraction.

analytical skills only apply to a circumscribed set of texts. One possible explanation for this effect is that since the scientific text is written to be taken literally, it only supports a small number of interpretations. If this were the case, one would expect there to be a high degree of overlap between the items that were classified as interpretations. In fact, 22 of the English student groups' 28 interpretive statements were unique. This suggests that although there was a large number of interpretations which could be derived from the scientific text, the English students were not facile at producing them.

These effects can be considered in more depth in the context of each of the texts. For each of the texts, whether the high school students' data is included or not, there is a significant interaction between group and the types of sentences, p's < .05. In all three texts, the English students generated the largest number of interpretations, the high school students generated the largest number of facts, and the engineers generated the largest number of other types of sentences. Both when all three groups, or only the graduate student groups are compared, there is a significant effect of group for both the short story and the poem, p's < .05, but not for the scientific text, p's > .10.

The content of the topic sentences. These analyses consider the content of subjects' proposed topic sentences. There were six possible categories: events, characters, setting, images, themes and language. A 3(text) x 3(group) x 6(content category) ANOVA was performed. Because these data overlap with those of the analyses of factual versus interpretive statements, only the main effect of content category and its interaction with other factors will be considered. These results are illustrated in Figure 3. The number of sentences varied across the content categories, $F(5,215) = 29.10$, $p < .0001$. Overall, the events and characters categories were addressed most frequently, at equivalent levels. Setting and imagery were written about least frequently. The distribution across content areas interacted significantly with group, $F(10,215) = 7.61$, $p < .0001$. The high school students concentrated especially on characters, whereas the English students wrote more about language than any other topic. The English graduate students also heavily emphasized themes, which were largely ignored by the other groups. The distribution across content areas also interacted with the kind of text, $F(10,430) = 35.08$, $p < .0001$. For the short story, the character category was used the most. For the poem, the language category was used the most. Lastly, in

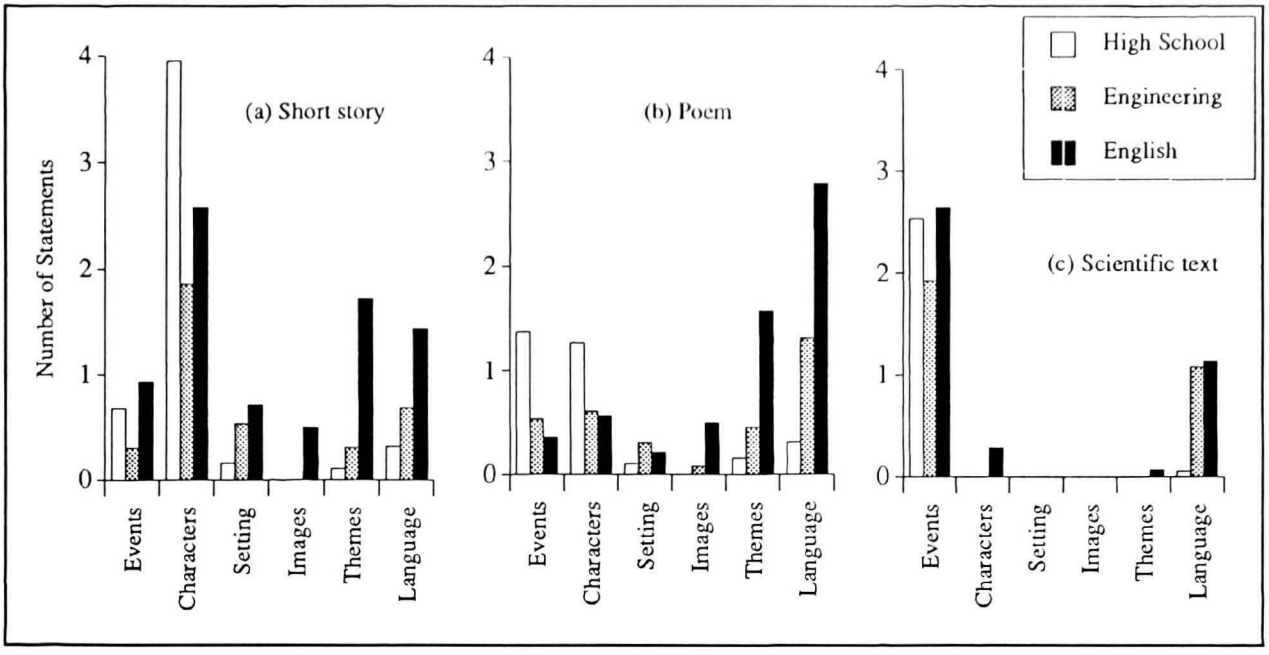


Figure 3. Number of topic sentences as a function of content category.

the scientific text, the events and language categories were used almost exclusively. Thus, on average, the subjects appear to note different aspects of each of the texts. The three-way interaction of content category, group and text also reached significance, $F(20,430) = 3.36$, $p < .0001$. The groups' patterns of category use differed for the poem and short story, but were nearly identical for the scientific text.

The same analysis was repeated without the high school students. The pattern of results was identical, with two exceptions. The interaction of group and content categories used did not reach significance, $F(5,125) = 1.39$, $p > .10$. While the English students wrote much more for each category, the ordering of their usage was similar to that of the engineers. The three-way interaction was marginally significant, $F(10,250) = 1.75$, $p < .10$. The patterns of category usage by the English and engineering students differed somewhat for the poem and short story, but they were practically identical for the scientific text.

When the texts were examined individually and the high school students' data were included, the interaction of group and aspect was significant for all three texts, p's $< .05$. When the graduate student groups are compared, this interaction only reaches significance for the poem, $p < .005$.

It is especially interesting that the English students did not write an exceptionally large number of topic sentences about the scientific text, even though they did so for the two literary texts. Notably, their production of interpretive statements is attenuated for this text. For the literary texts, the English students' extensive production of interpretations is indicative of a deeper level of understanding of these texts. They appeared quite capable of abstracting away from the facts presented in the literary texts. This supports the hypothesis that the English students' expertise is domain-specific. Their domain of expertise is not written language, but specifically literary texts.

Of the two literature texts, the poem is the more specialized genre. Narratives are more common reading fare. In the case of the poem, the aspects mentioned interacted with group, even when only the graduate student groups were considered. There was no significant interaction between content categories and group for the science

text when only the graduate students were considered. For this text the English students were at as much of a loss as the engineers. They were not capable of the more sophisticated and varied analyses that they performed in reference to the literary texts. In addition, this was the only text for which the English students did not produce many more interpretive than factual statements. Again, the domain-specificity of the English graduate students' analytic ability is apparent.

The text processing strategies of the English graduate students, seemed to be the most sensitive to the content and structure of each of the texts. Unlike the high school students, they did not focus on one dimension of each of the texts to the exclusion of all others. Also, the distribution of their comments across categories varies more between texts than the engineers'. They also use the more abstract end of the scale (images, themes and language) much more than the other two groups for the two literary texts. This is a clear example of their deeper analysis of these texts, and that they were able to go beyond the literal information stated. This claim is verified by the fact that the English students generated a higher ratio of interpretations to factual statements for the literary texts.

Conclusions

The results indicate that expertise in literature does exist, and that its general characteristics are like those of expertise in other domains. The English graduate students' performance on a variety of tasks justifies labeling them experts in literature. The implications of the expert-novice differences this study demonstrated are considered below.

These results refute the claims of literary theorists whose ideas are incompatible with the existence of domain-specific expertise in literature. The present findings indicate that experience in literature does lead to measurable advantages in processing literary texts. Some literary theorists would go further, suggesting that all one's experience with the world contributes to the processing of a literary text "by increasing the background knowledge brought to the task" (Fillion, 1981, p. 40). This implies that knowledgeable individuals can process literary texts in a more effective way than people with less knowledge. The results from the engineering graduate students refute this claim. Although they know a large amount about their own disciplines, their performance with respect to literary texts was often comparable to high school students and consistently inferior to the English graduate students'. Thus their large knowledge bases did not seem to aid their processing of literature as much as the literature experts' knowledge bases did. The results also contradict the claim of some literary theorists that experience in literature leads to advantages in other endeavors. The results indicate that the English students' expertise was limited to literary texts. The literature experts performed comparably to or worse than literature novices on a scientific text. The results suggest that the skills associated with expertise, even in ill-structured domains like literature, are domain-specific to a large extent.

Literary experts' knowledge bases. Literature experts' background knowledge allows them to focus on the appropriate level of detail. The memory results for literary texts indicate that the English students' representations differed from those of the other groups primarily at the gist level. Thus, the literature experts' meaning-level representation was more complete and accessible than the novices'. They suggest that literary experts have an additional kind of representation at a higher level of abstraction, that novices lack. This additional level is also evidenced in the experts' text analyses, in the way that they moved beyond the facts to interpretations of the texts.

Taken together, the results of this experiment offer strong evidence for the idea that literary experts have an additional, specialized level of representation, one step removed from concrete literal detail. By this argument, literary experts experience with literature has resulted in the development of a hierarchically organized knowledge

base with at least two levels. The first of these is a mundane literal detailed level of representation, which is comparable to that of novices. The second is a more abstract, principled level, which is the source of the expert-novice differences this study documents. The evidence suggests that experts must also have a certain facility in coordinating information from both levels. It may be the case that an additional, abstract level of representation may account for experts' superior abilities in other domains as well (e.g., Chi, Feltovich & Glaser, 1981; Dawson, Zeitz & Wright, 1989).

The domain-specificity of skills gained from the study of literature bears further investigation. This study demonstrated that literature experts' memory and analytic skills did not extend to another domain. However it remains to be resolved whether their superior analogical reasoning and argumentation skills (Zeitz, 1989) can be applied to other domains. In fact, although literature experts did not display an advantage in processing scientific texts, perhaps they would be superior to literature novices in processing information from domains more closely related to their own, such as history or another art form, such as painting. Clearly, further study of literature and other types of less structured domains can lead to a more complete understanding of the information processing abilities that develop with extensive experience in these types of domains.

Adelson, B. (1981). Problem solving and the development of abstract categories in programming languages. Memory and Cognition, 9, 422-433.

Chase, W. G., & Simon, H. A. (1973). Perception in chess. Cognitive Psychology, 4, 55-81.

Chi, M. T. H., Feltovich, P. J., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. Cognitive Science, 5, 121-152.

Dawson, V. L., Zeitz, C. M., & Wright, J. C. (1989). Expert-novice differences in person perception: Evidence of experts' sensitivities to the organization of behavior. Social Cognition, 7, 1-30.

Fillion, B. (1981). Reading as inquiry: An approach to literature learning. English Journal, 70, 39-45.

Glaser, R. (in press). Expert knowledge and processes of thinking. Chemtech.

Glaser, R. (1986). On the nature of expertise. In F. Klix, & H. Hagendorf (Eds.), Human memory and cognitive capabilities. Amsterdam: Elsevier Science Publishers.

Hatano, G., & Osawa, K. (1983). Digit memory of grand experts in abacus-derived mental calculation. Cognition, 15, 95-110.

Landow, G. P. (1989). Hypertext in literary education, criticism and scholarship. Computers and the Humanities, 23, 173-198.

Larkin, J., McDermott, J., Simon, D. P., & Simon, H. A. (1980). Expert and novice performance in solving physics problems. Science, 208, 1335-1342.

Mandel, B. J. (1976). What's at the bottom of literature? College English, 38, 250-262.

Purves, A. (1971). Evaluation of learning in literature. In B. S. Bloom, J. T. Hastings, & G. Madaus (Eds.), Handbook on formative and summative evaluation of student learning. New York: McGraw Hill.

Purves, A. (1984). The potential and real achievement of U.S. students in school reading. American Journal of Education, 93, 82-106.

Voss, J. F., Green, T. R., Post, T. A., & Penner, B. C. (1983). Problem-solving skills in the social sciences. The Psychology of Learning and Motivation: Advances in Research and Theory, 17, 165-213.

Zeitz, C. M. (1989). *Expert-novice differences in memory and analysis of literary and non-literary texts.* Unpublished doctoral dissertation, Brown University, Providence, RI.

This research was supported by the James S. McDonnell Foundation Program in Cognitive Studies for Educational Practice grant number 87-42 to Kathryn T. Spoehr and post-doctoral fellowship to Colleen Zeitz. I am grateful to Robert Glaser, Robert Hoffman, Gregory Murphy, Clark Quinn, Kathryn Spoehr, and Jacque Wright for helpful comments and suggestions on earlier drafts of this paper.

Learning from Indifferent Agents

LISA DENT LISA.DENT@CS.CMU.EDU
JEFFREY C. SCHLIMMER JEFF.SCHLIMMER@CS.CMU.EDU
School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213

(Received: March 14, 1990)

Abstract. In many situations, learners have the opportunity to observe other agents solving problems similar to their own. While not as favorable as learning from fully explained solutions, this has advantages over solving problems in isolation. In this paper we describe the general situation of *learning from indifferent agents* and outline a preliminary theory of how it may afford improved performance. Because one of our long-term goals is to improve educational methods, we identify a domain that allows us to observe humans learning from indifferent agents, and we summarize verbal protocol evidence indicating when and how humans learn.

Keywords: multi-agent domains, learning by doing, learning from examples, protocol analysis

1. Introduction

While getting out of my car, I saw another driver lock a bar-like object across his parked car's steering wheel, and I asked myself, "Why did he do that?" (As it turns out, it is a theft deterrence device designed to inhibit steering the car.) People often find themselves asking such questions, indicating that they build predictive models, apply them to other people, and attempt to explain surprising behavior. One benefit of this curiousity-driven behavior is that it can reveal novel strategies used by other agents; strategies that might otherwise be found only after extensive experimentation.

In more detail, let us consider three possible learning situations. In each, an agent has operators, a task to perform, and an indication of when the task is completed successfully; the agent must learn when to apply operators to complete the task. In the first learning situation, an agent learns in isolation, perhaps by trying operators and seeing what they do. This is called *learning by doing* (cf. Simon & Anzai, 1989), and the agent only sees outcomes that result from his/her actions. In the second situation, a teacher demonstrates successful and unsuccessful sequences of operators. This is called *learning from examples* (Carbonell, Michalski, & Mitchell, 1983), and the outcomes an agent sees are not biased by the current state of his/her learning. In the third situation, an agent learns with other agents that have similar operators and tasks but without a teacher. We call this *learning from indifferent agents* to denote our interest in situations where the agents are neither cooperative nor antagonistic. Here the agent has two sources of outcome observations: (a) those resulting from his/her own actions, and (b) those arising from the behavior of other agents. In this third situation the contexts surrounding observed agents' actions are likely to be underspecified because the reasoning of other agents may be based on unobservable factors.

Learning from indifferent agents involves a wide-spread class of situations that arise whenever there are multiple agents with comparable, but independent resources and tasks. Like the steering wheel example, people frequently have the opportunity to observe the behavior of

others without knowing exactly which factors lead to that behavior. This can happen because they are competing on a task or because the overhead of communication does not benefit the observed person. How do people make use of these observations? We hypothesize that learners may use the behavior of another agent for two ends: (a) as a source of novel, potentially useful operator applications ("Why did he do that?"), and (b) as a source of evaluation for current operator use ("She does that too"). We further hypothesize that learners will in fact make use of this information whenever they are able to adequately specify the situation surrounding the behavior. If too little is known about why the observed agent behaved as they did, then learners will be constrained to behave as if in a learning by doing situation. Conversely, if it is relatively easy to determine factors leading to the observed agent's behavior, then learners will learn higher quality knowledge more quickly.

2. A sample learning task

Card games are relatively simple task environments that afford opportunities for one agent to learn from another. Each player has comparable resources and goals. In some ways, card games relax unrealistic assumptions made in other games. Specifically, because players cannot see each other's cards and the cards are shuffled, card games do not involve the assumptions that agents have complete information or that the environment is completely deterministic. Chess, for instance, satisfies both of these assumptions, and Backgammon satisfies only the former.

As a domain of study, we have chosen the card game of *Mille Bornes* (1962, 1981) (or simply MB), which is modeled after a road race. Though the rules of the game are complex, and expert play is difficult, legal play in MB is relatively simple. Players begin with six cards, and on each turn they draw and then either discard or play a card. There are three types of cards: distance cards played to advance along the road (25, 50, 75, 100, and 200 miles), hazard cards played to temporarily stop an opponent (Flat Tire, Out of Gas, Accident, Stop, and Speed Limit), and remedy cards played to recover from hazards (Spare Tire, Gasoline, Repairs, Roll, and End of Limit, respectively). There is also a class of super-remedy cards called safeties that both remedy current hazards and protect from future ones (Puncture Proof tire, Extra Tank of gasoline, Driving Ace, and Right of Way, respectively). The primary object of the game is to travel either 700 or 1000 miles exactly, but the score depends on a number of factors, including: finishing first, using only distance cards less than 200, playing safeties, extending the play to 1000 miles, and keeping an opponent from playing any miles.

In this initial study, we chose two subjects who had played many card games before but had never played MB. They played MB against a computerized, expert player for about 1.5 hours each, their time consisting of three complete games of three or four hands each.[1] We asked the subjects to think aloud as they played, and we recorded their moves. One subject was instructed at the beginning of the second game to try to pay attention to the opponent's play and see what he could learn.

[1]Note that each play in a card game conveys information, but because subjects played against a computer implementation they correctly assumed that the opponent was neither trying to teach them nor trick them with its plays.

3. Evidence for learning from indifferent agents

While playing MB, subjects had the opportunity to learn by doing and to learn from the indifferent agent, i.e., the opponent. The subjects were free to choose if and when to use either technique. We are interested in how often learning from the opponent was used, in which situations it was used, and how it took place.

Learning from the opponent is possible when the opponent uses a strategy that the subject can infer from the opponent's overt action. In MB the only action that conveys information from the opponent to the subject is the opponent's move, usually a single card either played or discarded. In order to explain the reasons behind the opponent's move, the subject must deal with two unknowns: the cards in the opponent's hand and the set of strategies that the opponent is using to choose a move. One way of dealing with these multiple unknowns is to assume that the opponent has the same set of move strategies as oneself. Using this assumption and the opponent's move, aspects of the opponent's hand can be inferred. As long as the probability that the opponent has such a hand is high, one may conclude that the opponent is using a known strategy. If, however, the probability that the opponent has a hand that would lead to the observed move is low, one may consider alternative strategies that the opponent may be using.

For example, suppose the subject has a strategy to play the highest mile card available when able to play. If the opponent plays a 100 mile card, the subject can explain this move by hypothesizing that the highest mile card held by the opponent was a 100, which is very likely. However, if the opponent discards rather than playing a mile card even when able to play, the move can only be explained by hypothesizing that the opponent has no mile cards. Although this is possible, it is not common, so it may be useful to consider other strategies that could lead to this behavior. If the opponent is close to the 700 mile mark, one alternative explanation for the discard is that the opponent is saving mile cards until he has enough to reach 700 exactly.

Revising strategies in light of another agent's actions requires information about the context of those actions. If expert strategies require reasoning about information that is unobservable (e.g., contents of the opponent's hand), then it will be difficult for subjects to learn from their opponents. The converse is not in general true, though: if expert strategies require reasoning about observable information (e.g., cards already played), it may still be difficult for subjects to learn new strategies because of the relative difference between their strategies and optimal ones (see related work on expert/novice differences, e.g., Chi, Feltovich, & Galser, 1981).

Using this as an initial theory of the process of learning from an indifferent agent, we expect that subjects may learn from the opponent's behavior in two ways. When a move is explained by a strategy known to the subject, the subject may use this information as a positive evaluation of the strategy, because the opponent is known to be an expert player. When the move cannot be satisfactorily explained by current strategies, a new strategy may be learned if the context of the expert strategy is observable. Subjects learned from the opponent in both of these ways.

3.1 General observations

To test the initial theory just presented, we first examined the comments made by the subjects after each of the opponent's moves. Comment sequences referring to *why* the opponent made a particular move make up 3% of the total number of protocol statements, showing that the subjects are interested in the opponent's strategies and consider them a possible source of useful information. The majority of the remaining statements concerned the state of the game and the subject's own strategies.

Table 1. Learning from an indifferent agent: hypothesized factors and responses exhibited by subjects.

Strategy Name	Strategy Features		Type of Analysis			Total
	Observ.	Times Used	Confirm	Novel	Attempt	
Discard-useless	high	30	5	2	2	9
Extension	medium	9	1	0	1	2
End-limit	medium	4	0	1	0	1
Discard	low	195	1	1	6	8
Safety	low	26	0	0	0	0
End-game	low	13	0	0	0	0
Totals	—	277	7	4	9	20

We then categorized analyses made by the subjects into three groups. Analyses that explained the opponent's behavior in terms of a strategy currently used by the subject were classed *confirming*. Analyses that explained the opponent's behavior in terms of strategies not yet considered by the subject were classed *novel*. Finally, analyses that attempted to explain an opponent's move but were unable to do so satisfactorily were classed *attempted*. Specific examples of these types of statements appear in the next section.

According to our theory of the process of learning from the opponent, some strategies should be easier to infer from the opponent's move than others. Strategies with conditions that depend little on the cards in the opponent's hand will be more observable to the subject and therefore easier to identify. To investigate this hypothesis, we identified six, relatively complex strategies that subjects failed to use correctly at some point in the games. The six strategies are:

- Discard-useless. How to identify useless cards to discard.
- Extension. How to decide whether to extend the game to 1000 miles.
- End-limit. How to decide when to end a Speed Limit hazard.
- Discard. How to choose a card to discard when none are useless.
- Safety. How to decide when to play a safety card.
- End-game. How to decide which miles to play as the final mileage point (which must be reached exactly) is approaching.

Table 1 lists for each strategy: (a) the observability of the strategy, (b) the number of times the opponent used the strategy,[2] and (c) the number (by type) of analyses performed by subjects. At this level of detail, strategies are composed of a number of sub-strategies. We have initially encoded these using in a form similar to Siegler's (1976) encoding of strategies for the balance scale task. Each analysis made by the subject pertains to a particular sub-strategy. Because the observability of the strategy varies from sub-strategy to sub-strategy, Table 1 lists an overall qualitative assessment of observability.

[2]The number of times a strategy is used is approximated by applying a rational model to the observed agent's behavior; exact information is unavailable because the observed agent's hand is unknown.

The table shows that in general the observability of a strategy does increase (a) the number of analyses attempted by the subjects, and (b) the number of cases where a novel strategy is learned. For the high observability Discard-useless strategy, subjects display considerably more analyses than for the low observability Safety strategy along both of these dimensions (total analyses 9 to 0, novel analyses 2 to 0). However, the Discard strategy does not follow this pattern. Although it is a low observability strategy, subjects still analyze it eight times. Another factor appears to be operating here: the frequency of exposure to a strategy. As the table shows, the Discard strategy is used by the opponent extremely often. Perhaps with repeated exposure to the aspects of the strategy which are observable, the opponent's hand can be successfully inferred. This frequency must be very high to overcome the hindrance of low observability; subjects give no analyses of the low observability Safety strategy in spite of the fact that it occurs three times as often as the medium observability strategy Extension, which receives two analyses. Another potentially relevant factor, which we have not considered, is the complexity of the strategy compared to the subject's skill level.

We now turn to a specific class of strategies to illustrate the process of learning from indifferent agents in more detail.

3.2 A case study: The discard strategies

To supplement the high-level, quantitative analysis in the previous section, in this section we focus on one subject's specific analyses of his opponent's discard behavior. Because the discard action is a part of many card games, a person familiar with other card games (e.g., Gin Rummy) brings discard strategies from these games to MB. These become the initial set of discard strategies, which are modified during play as the person learns about the unique features of MB. Studying the protocols reveals that the subjects' simple initial discard strategies are:

1. Discard useless cards first.
2. If no useless cards, discard card of least expected utility:
   (a) Discard duplicate cards by rank.
   (b) Discard duplicates of a category by rank.
   (c) Discard singles by rank.

These initial strategies are expanded and modified by the subjects as the game progresses. The opponent's fixed set of discard strategies corresponds to the fully expanded forms of these two strategies.

The first strategy, the Discard Useless (DU) strategy, requires the identification of useless cards. Because this feature is unique to MB, the subjects must learn it as they play. There are several cases in which a card becomes useless. These are listed below along with the game and hand in which the strategy is first used by a subject, if at all. Note that most of the cases do not require information about the opponent's hand and thus are highly observable. The opponent uses a complete version of the DU strategy throughout the games.

1. If the player has played a safety card, the corresponding remedy is useless (Game 1, Hand 1).
2. If the opponent has played a safety card, the corresponding hazard is useless (Game 1, Hand 1).

3. If all hazard cards of a particular type have been played, the corresponding remedy is useless (not used).

4. If the player has played two 200 cards, more 200 cards are useless (not used).

5. If the player is within 50 miles of his final point, the End of Limit (EOL) card is useless (Game 3, Hand 2).

6. If the opponent is within 50 miles of his final point, the Speed Limit (SL) card is useless (Game 3, Hand 2).

7. If the player is approaching a final point, mileage cards that put the player over that point are useless (Game 3, Hand 2).

Because the DU strategy is highly observable and often used by the opponent, we expect that subjects will be able to acquire it by observing the opponent's actions. One subject first identifies an instance of a useless remedy by observing the relationship among his own cards. When he needed to discard for the first time, he used Case 1 of the DU strategy. Later in the same hand, the opponent discarded a useless remedy, and the subject was quick to notice and explain this behavior, perhaps confirming his own strategy in the process (Game 1, Hand 1, *confirming*):

> He discarded a Gasoline, obviously, cause he's got an Extra Tank.

Thus the subject acquired the first case of the DU strategy by doing rather than from the opponent, using the opponent's moves only to confirm the strategy.

An interesting example of learning a novel DU strategy from the opponent occurred in the subject's acquisition of the 5th and 6th cases of the strategy. When the subject had 675 miles, he observed the opponent discard a Speed Limit and attempted to explain the opponent's behavior using the DU strategy (Game 3, Hand 2, *novel*):

> He discarded a Speed Limit. Wonder why he did that? Oh, because he knows that I only need a 25 to win is that why?

The subject inferred that the opponent discarded the Speed Limit because it was a useless card. Another possible explanation, and in fact the correct one, is that the opponent had two Speed Limit cards. The subject then deduced a version of the fifth instance of the DU strategy:

> So I get rid of the End of Limits obviously.

However, these versions of the DU strategy are slightly incorrect. They test whether the player is within 50 miles of 700, not whether the player is within 50 miles of the *final point*, which may be 1000 miles if the player elects to extend the game. In fact, the subject did go on to extend this game, and after doing so discovered his earlier error.

This episode illustrates some of the pitfalls of learning from indifferent agents. It seems likely that strategies with complex conditions are prone to such errors (especially if the strategies are completely new to the subject). However this episode also shows that the opponent's moves can be used to acquire strategies that might otherwise be missed.

The DU strategy is highly observable. If a play is not possible, then its use does not depend on other cards in the hand. This is not true of the second initial strategy (discard the card of least expected utility), making it more difficult to learn by observing the opponent.

However, after observing the opponent discard a 25 mile card, the subject is successful at modifying a discard strategy. Up to this point the subject had been ranking all mile cards above remedy and hazard cards. He noticed the opponent discarding a 25, and considered the possibility of changing his ranking of low mileage cards in certain situations (Game 2, Hand 1, *novel*):

> He had to discard, which is good. He discarded his lowest point value. So it seems like one idea is to ...25s don't seem that important, especially if you have two of them like I do ...because you may need a 25 to finish ...to get exactly to 700, but ...

More often, however, a subject remarks on the opponent's discard and gives an unsatisfactory explanation, unable to infer the correct strategy from the observation (Game 2, Hand 1, *attempted*):

> He just discarded a Roll Card, boy, why did he do that? He must be pretty desperate to discard a Roll Card. He must have a lot of those.

These specific examples of learning discard strategies from observing the behavior of the opponent illustrate two important features of our theory. First, subjects attempt to explain opponent's moves first in terms of their own strategies and then, if necessary, in terms of novel strategies. Second, they are more often successful in their explanation when the conditions of the strategy are highly observable. Other factors influencing their success include the number of opportunities to observe the strategy in use and the complexity of the strategy itself.

4. Conclusions

Whether playing a game or living their daily lives, people pay attention to the behavior of other agents, and in some cases, they learn from those observations. In terms of the information that the learner sees, the task of learning from indifferent agents is a composite of learning by doing, where the learner generates learning opportunities, and learning from examples, where another agent generates learning opportunities.

Our initial theory postulates a specific relationship between the nature of an observed strategy and the ease of learning it from an indifferent agent: the less observable the factors measured by the strategy, the more difficult it will be to learn. This has a direct application for the design of instructional situations. In a field-classroom setting, to facilitate student learning by watching an expert perform a task, ensure that the factors weighted by the expert are clearly identified for students. It may not be as important to identify the expert's interpretation of those factors explicitly. Applying this principle to the card game we studied here, we hypothesize that the failure to learn one of the low observability strategies (e.g., Safety) may be facilitated by the inclusion of even small amounts of additional information (e.g., the number of safeties that the opponent has).

Our initial theory is far from a complete theory predicting the efficacy of learning from an indifferent agent. The protocols reveal that subjects do not always learn from their opponent even when the optimal strategy measures only observable features, that they do not consistently apply strategies they have verbalized, and that they do not consistently notice behavior that is inconsistent with their verbalized strategies. A more complete theory would require accounting for the semantic distance between the learner and the observed agent's strategies, practice effects, and attentional mechanisms. If an accurate and comprehensive theory could

be formulated, it would have a significant impact on the design of instructional materials and artificial learning methods.

Acknowledgements

We would like to thank Randy Jones for his helpful suggestions and careful reading of an earlier draft of this paper. This research is supported by the National Science Foundation under grant IRI-8740522 and by a grant from Digital Equipment Corporation.

References

Carbonell, J. G., Michalski, R. S., & Mitchell, T. M. (1983). An overview of machine learning. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach.* San Mateo, CA: Morgan Kaufmann.

Chi, M. T. H., Feltovich, P. J., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science*, *5*, 121–152.

Mille Bornes rules (1962, 1981). Beverly, MA: Parker Brothers.

Siegler, R. S. (1976). Three aspects of cognitive development. *Cognitive Psychology*, *8*, 481–520.

Simon, H., & Anzai, Y. (1989). Theory of learning by doing. In *Models of thought.* Yale University Press.

Using Knowledge Representation to Study Conceptual Change in Students for Teaching Physics

C. Franklin Boyle, CDEC, Carnegie Mellon University, Pittsburgh, PA 15213
Dewey I. Dykstra, Jr., Department of Physics, Boise State University, Boise, ID 83725
Ira A. Monarch, Department of Philosophy, Carnegie Mellon University, Pittsburgh, PA 15213

Abstract

Our goal is to understand the development of physics concepts in students. We take the perspective that individuals construct their own understanding so as to 'fit' their experiences. This constructive activity results in conceptions about the physical world. The major challenges in physics instruction then are the tasks of identifying and inducing change in students' conceptions about the physical world. Our efforts to understand the nature of conceptual change are aided by knowledge representation techniques. We present examples in which some of the finer structure of conceptual change is represented which illustrate the potential of knowledge representation for studying conceptual change.

Introduction

Studies have found that students begin formal physics education with a system of physical *conceptions* that differ in deeply systematic ways from those of the physicist and present a significant obstacle to learning physics (Viennot, 1979; di Sessa, 1983; McDermott, 1984; Halloun and Hestenes, 1987). These *alternative* conceptions manifest themselves as *useful* commonsense beliefs about the world. Yet alternative conceptions are not addressed by standard instruction, either in physics classrooms or in introductory physics textbooks. Simply presenting students with the laws of Newtonian mechanics when they solve problems, for example, does not encourage conceptual learning because such statements do not make the intended sense in the context of their beliefs; instead the students interpret what is said in terms of their *existing* beliefs. As a result, most students depart from beginning physics courses without an understanding of the Newtonian *concepts* presented; their conceptual framework about how the world works is left essentially unchanged.[1] If physics instruction is to encourage the kind of learning which leads to new conceptual understanding, it must address the alternative conceptions that need to be changed. In our view, such conceptual change occurs only when new conceptions are *constructed* by individuals themselves. We believe students can make sense of Newtonian concepts if they experience situations which bring them to question their own conceptions and are then facilitated in their attempts to develop more viable replacements.

Identifying alternative conceptions has been the focus of a number of studies which have provided qualitative analyses of many of the conceptual difficulties students have in beginning physics (Minstrell and Stimpson, 1986; McClosky, 1983; McDermott, 1984; and Clement, 1982). The results of these analyses and others', as well as our own, have revealed many of the alternative conceptions associated with beginning mechanics. These results need to be integrated and organized so they can be used to study necessary conceptual changes in learning physics. We believe this requires representing alternative conceptions as structures comprised of salient features and relationships. We call these representations "conceptual maps". They represent in an explicit and pragmatic way, concepts, terms, features and their interrelationships that underly students' descriptions of the physical world. Successive conceptual maps represent in detail the changes students go through from "motion implies force", for example, to the Newtonian conception that "acceleration implies force". Such representations aid in identifying desired conceptual changes which can be associated with instructional techniques found to be effective for bringing about conceptual change. The adequacy of the maps is based ultimately on how well their representation of conceptions can effectively inform instruction.

Alternative Conceptions

Alternative conceptions make up students' fundamental beliefs about how the world works or how it is constituted which are quite different from those of the physicist. These beliefs apply to a variety of different

[1] This does not mean that these students necessarily get poor grades. Students are still able to solve the problems assigned without understanding of the underlying conceptions (Halloun and Hestenes, 1987).

situations. They are beliefs in an *explanatory* sense about causality. For example "motion implies force"[2] is a conception which leads students to suggest that,

a.) there is a force (in addition to gravity) that propels a block down an inclined plane because it moves down the plane *or*
b.) there is a force in the direction of a planet's motion *or*
c.) there is a force down on an object because it can move down.

Another example is the "materialistic conception" which leads students to endow things like electricity, light, and heat with matter-like properties (Reiner, Chi, Resnick, 1988).

Identifying Alternative Conceptions

In contrast to modeling problem-solving skills, studying conceptions is more difficult because there is a "level of indirection" between what we observe students doing and the conceptions that give rise to their behaviors. Conceptions must be *applied* to problem situations and thus can only be identified indirectly by analyzing student responses. This is in contrast to skills which are specified in terms of observable features of the problem situation (Anderson, Boyle, Corbett and Lewis, 1990) independently of what students may think.

In order to identify students' conceptions we must be able to make sense of their behaviors. We do this by applying a model of conceptual application, depicted in Figure 1, which is based on the hypothesis that when conceptions are applied in specific problem contexts, they are manifest as characteristic behaviors. For example, when two interacting objects have different sizes, students associate a larger force with the larger object. We would expect to see similar behaviors resulting from such a belief in different problem situations sharing this feature. But to be more convinced that we identify the underlying conception, we consider problem situations that include additional features, that is, we consider a variety of situations in which that conception might be applicable. This is because the specific

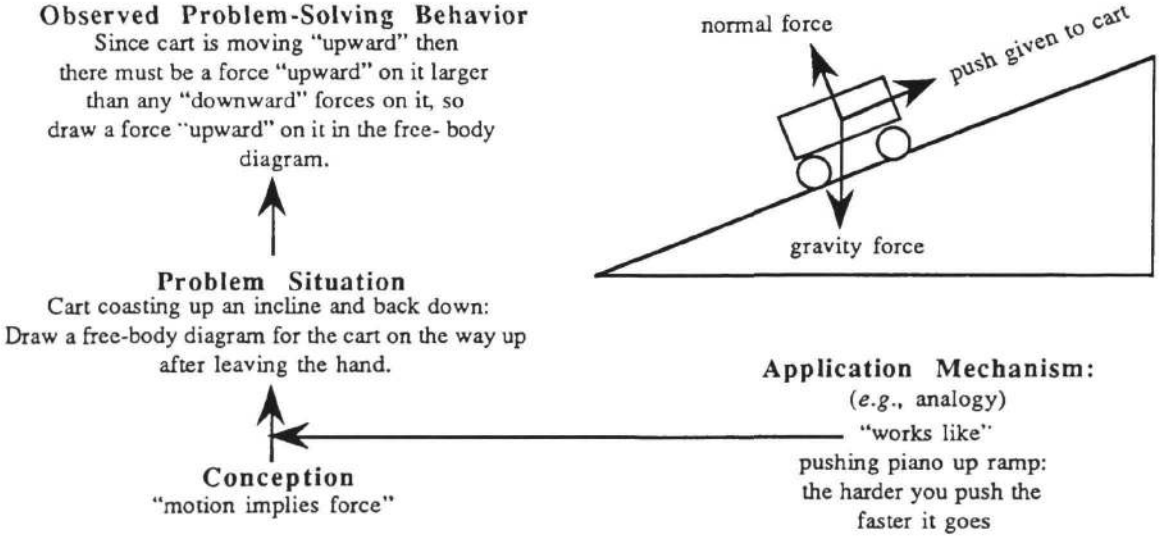


Figure 1: Model for Conceptual Application
A conception applied to a problem situation results in an observable problem-solving behavior. Note that in this example there is also evidence of force being "given to an object as if it were a property of the object.

behavior of associating a larger force with the larger of two interacting objects is probably the result of applying a conception like "force is a property of an object" in addition to a *specific* belief about different sized objects. By having students solve a number of different kinds of problems, we work backwards to determine whether a more

---

[2] We are aware that on some interpretations "motion implies force" is even true for Newton, *i.e.* the persistence of motion implies an internal force or *vis insita* (McGuire,1990). However, the alternative conception we mean by "motion implies force" is different from this interpretation of Newton in that for students "force" has *not* been differentiated into 'force of persistence' and 'force of acceleration', although we would not be surprised to find the students making this same differentiation as they begin to form more Newtonian conceptions.

general conception underlies the belief. For example, students apply "motion implies force" to a wagon pulled by a donkey. But they also apply it to a body moving *without* apparent forces which is a very different kind of situation. This leads students to remark that the body is moving because force has been transferred to it from a body that originally set it in motion (similar to impetus).

The conceptual application model serves as a methodological framework for identifying conceptions. It is also explanatory of what we observe, enabling us to make sense of students' behavior. However it is not a detailed psychological model in that it makes no claim as to the real mental or physical nature of conceptions nor does it specify the mechanisms by which they are applied. Rather, it is a model in the sense of a concrete construction which is used to catalogue, in a principled and organized way, the kinds of descriptions that students give in answers to questions and that can be inferred from their behavior in problem-solving situations.

Conceptual Change

Conceptual change has come to play an important role in the history and philosophy of science in the last several decades (Kuhn, 1970; Feyerabend, 1988). These studies have begun to influence research in cognitive developmental psychology (Carey, 1985). Carey describes two possible senses of knowledge restructuring, *weak* and *strong*. According to Carey, knowledge restructuring in the weak sense involves a rearrangement of the relationships between existing *concepts* such as velocity, acceleration and force and the situations to which they apply. In weak restructuring, concepts are not changed, rather their applications are either extended, restricted, or rearranged. Knowledge restructuring in the strong sense involves *changes* in the concepts themselves, *i.e.* actual conceptual change. We agree with Carey's provisos on conceptual change with some additional refinements that are a result of explicitly representing knowledge in conceptual maps.

An Example of Conceptual Change

The following is a description of our observations of a sequence of conceptual changes involving the relationship between force and motion, depicted in Figure 2, along with the method of inducing those changes (Minstrell, 1989). The method was originated by Minstrell almost ten years ago.

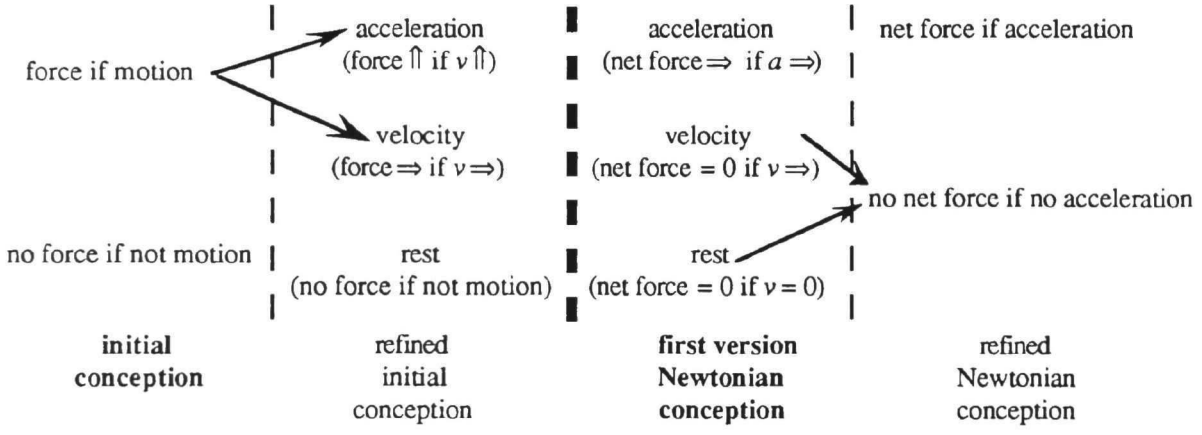


Figure 2: A Series of Conceptual Changes
The bold, vertical, dashed line at the center of the figure indicates a substantial conceptual change. The regular, vertical, dashed lines at either side indicate less substantial conceptual refinements ("⇑" -- increases; "⇒" -- remains constant).

Students typically come to us at the introductory college level with the conception, "motion implies force", together with an undifferentiated view of motion. Even if students can *recite* Newton's three laws of motion, their response to questioning and their problem-solving performance usually reveal conceptions which are not in accord with their statement of the laws. This situation is indicated by the leftmost column in Figure 2, labeled "initial conception". Following a series of laboratory-based activities using microcomputer-based laboratory (MBL) equipment (Thornton, 1987), students figure out the interrelationships between various quantities used to describe the motion of particular objects from graphs of distance *vs.* time, velocity *vs.* time, and acceleration *vs.* time. They begin to discriminate different motions when reporting about falling bodies. For example, they refer to "speeding up" or "acceleration" instead of just "falling down". This captures a distinction between the first and second columns in Figure 2.

This differentiation of motion does not, by itself, change students' conceptions about the *causes* of motion, but it does generate a conceptual division of motion. With respect to causation, this change provides a basis for elaborating or enriching their current conception. For example, students will say that to maintain a constant velocity a constant excess force is needed and if there is a changing velocity, then the excess force is changing as well. This situation is depicted in the second column from the left in Figure 2, labeled, "refined initial conception".

After students have made this distinction explicitly in class, they are asked to *observe* the motion of an object which is clearly under the influence of a constant excess force and *are surprised* to find that it does not move with constant velocity, but with constant acceleration. Their surprise is partly indicative of a cognitive *disequilibration* (See below). Upon further reflection, they realize that this invalidates their explanation for constant velocity which is probably now best represented by "velocity implies force". In classroom situations, where the teaching method is constructivist, discussion[3] is encouraged in which students come to realize that a zero magnitude excess force is a condition for constant velocity which is more consistent with their new observation about acceleration. In arriving at this view they also shift their consideration of force from applied force to what the physicist calls *net force*. This conceptual change is represented as a shift from "refined initial conception" to "first version Newtonian conception" in Figure 2.

This series of laboratory-based activities leaves students short of what we would call a "refined Newtonian conception" in that the equivalence of constant velocity and rest has not been completely resolved. It is currently our hypothesis that differentiating between rest and zero instantaneous velocity underlies the change to a "refined Newtonian conception". While this differentiation is addressed in a purely kinematic context for the coin-toss problem during the students' MBL work, it should be brought up in the context of applying their new conception (first version Newtonian conception) to acceleration during a coin-toss. Another context in which the equivalence between constant velocity and rest might be addressed is in the consideration of inertial reference frames. This differentiation might clear the way for understanding *rest* as a particular state of constant velocity.

Assimilation, Accommodation and Disequilibration

Changes in the knowledge state of a learner are often described as assimilations and accommodations. These terms were initially introduced in the context of learning by Piaget. We introduce assimilation, accommodation and especially disequilibration into our discussion of conceptual change in order to describe, from the standpoint of learning and pedagogy, the necessary conditions for conceptual change. While we think of conceptual change in these Piagetian constructivist terms, we *do not* invoke the notion of Piagetian stages of cognitive development.

Assimilation is the recognition that an event (physical or mental) fits an existing conception (von Glasersfeld, 1987). This recognition process also involves a selective ignoring of discrepancies deemed not salient. Accommodation is a change in fundamental belief about how the world works, that is, a change in a *conception*. It is the construction of a new structure which *can* assimilate an event which could not be assimilated under previously held conceptions. Each of the three transitions in Figure 2 involves accommodation. Where the initial conception becomes refined and where the first version Newtonian conception becomes refined, we would say that "within conception" accommodation has occurred. In the former, for example, motion is differentiated into velocity and acceleration, but the conception that "motion implies force" remains essentially intact. "Conception-change" accommodation occurs at the vertical, bold, dashed line; *i.e.*, where the conception changes from the initial, everyday conception to a more Newtonian conception. The difference between "within conception" and "conception-change" accommodation is a difference in how fundamental the change in the students' knowledge structure is. It is not the same as the difference between Carey's weak and strong knowledge restructuring.

We believe that for accommodation to occur, the learner must become motivated to change by passing through a state of cognitive disequilibration; sometimes profound, sometimes not, but always a disequilibration. When an individual cannot fit an event into existing beliefs, a state of disequilibration exists. The fact that certain conceptions may not change under normal instruction may be due to the failure of that instruction to disequilibrate students with respect to the conceptions they hold. If students can assimilate events (words, ideas, experiences) presented in the course of instruction, then there is no disequilibration and no conceptual change. The point of instruction should be to induce conceptual change. It cannot accomplish this without *inducing* disequilibration and *facilitating* accommodation.

It should be noted that disequilibration is *not* contradiction. The latter refers to a logical inconsistency whereas disequilibration is a conceptual incongruity. Disequilibration is not a consequence of formal, truth-valued statements, but, rather, the surprise produced when an expected physical event does not occur. Conceptual change does not depend on contradiction, but on disequilibration. [4]

---

[3] It is important that this be a discussion between students, not between students and teacher.

[4] For example, while people who believe in heliocentrism would find the retrograde motion of the planets a contradiction of geocentric beliefs this was not the case for those who held the geocentric view. For them, the retrograde motion of the

Representing Conceptual Knowledge for Instruction

A knowledge representation scheme that we believe can capture the kinds of differences illustrated in Figure 2 includes a subcategorization hierarchy (Korner, 1970) represented as a network of nodes and links between them which can be instantiated in a computer (Brachman, 1979; Nirenburg and Monarch *et al.*, 1988), as depicted in Figure 3.[5] The nodes at the highest level represent the most abstract categories such as object, attribute, relation and situation (*e.g.* state, process), becoming less abstract traveling down the hierarchy. Links are relations between categories. In the domain of mechanics, the primary kind (*i.e.* subcategory) of object considered is "physical object" and example kinds of "situation" are "motion", "constant velocity" and "accelerated motion" (the latter two are subcategories of "motion"). Examples of

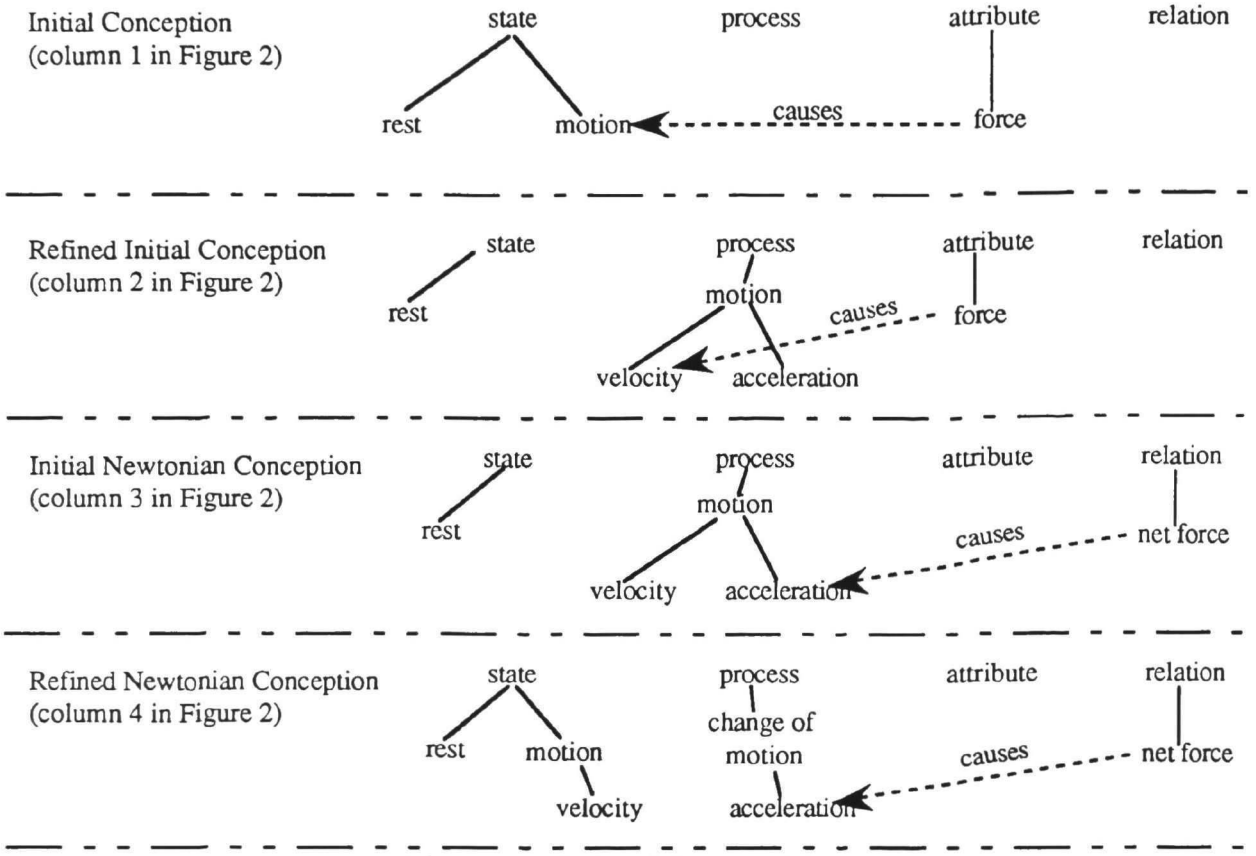


Figure 3: A Series of Conceptual Maps
A network representation of the force and motion conceptual changes which are depicted in Figure 2.

relations are "takes place in" and "causes", while "mass'" is an example of an attribute. The four networks in Figure

---

planets was explainable by epicycles and thus not seen as contradicting the belief that all heavenly bodies revolve around the earth. While the identification of retrograde motion did not contradict geocentric beliefs, it did create enough cognitive dissonance to require the introduction of epicycles, *i.e.* a "within-conception" accommodation. When epicycles became too unwieldy, this contributed, along with other factors, to further disequilibration and finally to "the Copernican Revolution". Even more telling of the difference between contradiction and disequilibration is that what appears a contradictory event to some may not disequilibrate others. This frequently happens in traditional instruction. Carefully planned activities fail to lead students to the conclusions teachers intend. Students frequently fail to be disequilibrated by the experience of seeing a material at constant temperature while its surroundings change temperature during heating and cooling through state changes.

5 It should be noted that the diagrams in Figure 3 are schematic, showing only enough detail to illustrate our points. Considerably more detail, we believe, is necessary for capturing students' knowledge relating to force and motion for instructional purposes.

3 correspond to the columns in Figure 2. In the first network there is no subcategorization of "motion" and "force" is causally linked with this undifferentiated "motion". In the second network, "force" is causally linked to "velocity" which is now a part of a differentiated "motion." In the third network the notion of force has become "net force", a relation between objects instead of an attribute of an object and it is causally linked to "acceleration." In the fourth network "motion" becomes a state like "rest" while "change of motion" is a process -- thus "velocity" is "motion" as a state while "acceleration" is a change of motion.

Conceptual Maps

We call a network representation of the knowledge state of a student a "conceptual map". It enables us to give precise and explicit specifications of the elements and relationships associated with conceptual change. The nodes in the network represent what we call individual *concepts* (*e.g.* "force") while conceptions can be expressed as propositions (*e.g.* "motion implies force") which are combinations of concepts. The proposition, "motion implies force", indicates a causal relationship between the concepts of force and motion that is explicitly captured in the network. Though conceptions are mental entities, we are able to *refer* to them through propositions represented explicitly in the map. Since conceptions, like "motion implies force", are the basis for understanding how the world works, they are represented by (causal) relationships between nodes relatively high up in the network hierarchy. Conceptual change in general gives rise to different kinds of changes in the network. For example, a conception may be "enriched" by the differentiation of an existing concept that is part of the conception. This differentiation is represented as the generation of subcategories in conceptual maps (within-conception accommodation). For example, in the first column of Figure 2, motion is undifferentiated, yet in the second column it becomes differentiated leading to an enriched conception.

Our scheme for representing students' knowledge is unique in that it enables us to bring together a constructivist (Glasersfeld, 1984) learning framework and knowledge representation techniques. The representation makes explicit the categories by which students discriminate different situations and the conceptual apparatus they use to identify and describe those situations. There are recent indications that others interested in science education are beginning to use knowledge representation techniques (Nersessian, 1989). Skill models give a precise way of modeling the student's skill knowledge, and can be tested by model tracing, which involves simulating problem-solving behavior. Likewise, conceptual maps will provide a concise way of modeling conceptual knowledge since they can be used to trace a student's conceptual learning. Accommodation can be represented as a sequence of two or more conceptual maps tracing a conceptual change in which conceptions are altered either by enriching them through new subcategories or changing relationships between categories. Thus, an accommodation can be tracked going from a network representing motion as a process requiring force to a more Newtonian network in which acceleration is a process requiring force.[6]

There are two important points to note here. The first concerns abstract categories like "process" and "state", as depicted in Figure 3, while the other concerns the problem 'tracking' concepts across conceptual changes. With regard to the first point, we are not recommending that beginning students learn to distinguish the differences between certain abstract categories, like "process" and "state", and to be able to classify certain physical concepts under them. What we *are* recommending is that teachers recognize the categorial affiliations of student concepts so that they can better understand their students' learning needs.

As regards the second point, we believe that overlaps in the situations to which the concepts are applied and the stability of many of the concepts from network to network will allow us to identify earlier and later versions of the same concepts. While this view regarding the traceability of conceptual change is similar to others (Carey, 1985; Keil, 1989), we believe our conceptual maps, or networks, will specify much more explicitly what changes and what remains the same in conceptual change. Conceptual networks will not only represent alternative conceptions but Newtonian conceptions as well. We will be able to represent all manner of intermediate conceptions in different maps from a student's initial alternative conceptions to the aimed for Newtonian conceptions.

The Knowledge Framework

The framework for representing knowledge encompasses conceptual maps for pedagogically relevant knowledge states and for the set of conceptual changes which link them. The maps are built by analyzing data, acquired from student interviews and problem-solving tasks, into conceptions and category distinctions or concepts. Representing this analyzed data involves the identification of pedagogically relevant knowledge by an examination of

---

[6] In a Newtonian framework, motion as constant velocity is a *state* like rest. The preservation of such states do not require external forces (McGuire, 1990). Only accelerated motion as a change of motion or change of state is a *process* requiring external force. Such "conception-change" accommodation may require changes in ontological categorization (Korner, 1970). In this case the conceptual change is represented in part as a differentiation of motion, where the differentiated categories are each classified under different ontological categories, process on the one hand, state on the other (Koyre, 1968; Kuhn, 1970; Feyerabend, 1988)

student conceptions in comparison with Newtonian conceptions. As is illustrated in Figures 3, some of the salient differences are causal linkages and hierarchical relations of physics concepts to various ontological categories. Conceptual changes occur in the context of knowledge elements that appear to be invariant from student to student.[7] These invariant elements include abstract ontological categories like "state" and "process" on the one hand and concrete elements such as prototypical scenarios which people have in common (for example, students typically refer to the relation between the "gas" pedal in a car and the motion of the car). These concrete knowledge elements are very important because they are what students often argue from to support their pre-Newtonian conceptions. Having students carefully examine these elements can be the basis for students to begin conceptual change (Minstrell, 1989).

Thus, conceptual maps are comprised of ontological categories and concrete elements, as well as physics concepts such as "motion", "velocity", "force", *etc.* Conceptions (both alternative and physicists') which can change are situated against the background of the stable framework of ontological and concrete elements. The fact that there are stable elements in a conceptual map constrains the number of possible variations and makes the task of mapping a student's conceptions less difficult than might be imagined.

Once conceptual maps of student ideas have been built, desired conceptual changes must be determined. Such maps together with maps of goal conceptions can be used to draw connections between alternative conceptions and goal conceptions, for example, implied by the sequence of conceptual changes depicted in Figure 3. These *may* reveal whether there are other knowledge-state elements or structures which must be further divided as preconditions as these conceptual changes, such as differentiating the concept of motion described above.[8]

Conclusion

Pedagogically, conceptual maps enable instruction to focus on explicitly depicted aspects of students' understanding through the kinds of distinctions students make when they think about the physical world. That is, the maps organize and make explicit the essential content of experiences intended to disequilibrate students. Thus, analogous to problem features that comprise the conditional sides of rules used to model skills associated with specific problem-solving actions, portions of conceptual maps can be considered to represent features of student knowledge states associated with conceptual change techniques that effect changes in those knowledge states. Conceptual maps, as we have described them, enable us to monitor and evaluate student learning by providing a more precise way of representing conceptual change. The more salient detail that is included in the networks, the more accurate will be the representations of students' knowledge on which instruction will be based and, we believe, the more effective will be instruction.

References

Anderson, J.R., Boyle, C.F., Corbett, A. and Lewis, M. Cognitive Modeling and Intelligent Tutoring, *Artificial Intelligence* **42**, 7(1990)

Brachman, R. On the Epistemological Status of Semantic Networks. *Associative Networks: Representation and Use of Knowledge by Computers.* Findler, N. V. (ed) Academic Press, New York (1979).

Carey, S. *Conceptual Change in Childhood.* The MIT Press, Cambridge, MA (1985).

Clement, J. Students Preconceptions in Introductory Mechanics. *American Journal of Physics* **50**(1), 66(1982)

di Sessa, A.A., Phenomenology and the Evolution of Intuition. *Mental Models*, Gentner, D. and Stevens, A.L. (Eds.), Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey, 15(1983)

Feyerabend, P. *Against Method, Revised Edition*, New York: Verso, (1988).

Glasersfeld, E. v. An Introduction to Radical Constructivism. in *The Invented Reality*, Watzlawick, P. (ed.), New York: W. W. Norton & Co., Inc., 17(1984)

Glasersfeld, E. v. Cybernetics, Experience, and the Concept of Self. *The Construction of Knowledge: Contributions to Conceptual Semantics.* A volume of articles written by Ernst von Glasersfeld, The Systems Inquiry Press, Salinas, CA, (1987).

Halloun, I.A. and Hestenes, D. Modeling Instruction in Mechanics. *American Journal of Physics* **55**(5), 455(1987)

Keil, F.C. *Concepts, Kinds and Cognitive Development.* A Bradford Book, MIT Press, Cambridge, MA (1985).

Korner, S. *Categorial Frameworks.* Barnes & Noble, New York, (1970).

---

[7] For other views which share the intuition of this assumption see Smith *et. al.* (1985) and Keil (1989).

[8] Often it is the case that new forms of representation can reveal unanticipated, valuable insights. Boyle (private communication) has indicated that representing the skills necessary for solving proof problems in geometry revealed a previously unnoticed proof in a pedagogically important geometry problem. Similar surprises have occurred in physics where new phenomena have been predicted from new representations, *i.e.* new particles predicted by applying group theoretical representations.

Koyre, A. *Metaphysics and Measurements.* Harvard University Press, Cambridge, MA (1968)
Kuhn, T. *The Structure of Scientific Revolutions.* University of Chicago Press, Chicago, IL (1970)
McCloskey, M. and Kohl, D. Naive Physics: The Curvilinear Impetus Principle and Its Role in Interactions with Moving Objects. *Journal of Experimental Psychology: Learning, Memory, and Cognition* **9**, 146(1983)
McDermott, L.C. Research on Conceptual Understanding in Mechanics. *Physics Today* **37**(7), 24(1984)
McGuire, J. E. Newton on *vis insita* of Bodies. Conference on Self-Motion: From Aristotle to Newton, University of Pittsburgh, Pittsburgh, PA (February, 1990).
Minstrell, J. A. Teaching Science for Understanding. *Toward the Thinking Curriculum: Current Cognitive Research*, Resnick, L. and Klopfer, L. (eds), Alexandria, VA, Association for Supervision and Curriculum Development, 129 (1989).
Minstrell, J. and Stimpson, V. Instruction for Understanding: A Cognitive Process Framework. Final Report for NIE Grant G 83-0059, U.S. Department of Education, Washington, DC, (1986)
Minstrell, J. Explaining the 'At Rest' Condition of an Object. *The Physics Teacher* **20**(1), 10(1982)
Nersessian, N. J. Conceptual Change in Science and Science Education. *Synthese* **80**, 163(1989).
Nirenburg, S. and Monarch, I. A. Acquisition of Very Large Knowledge Bases, Technical Report, Center for Machine Translation, Carnegie Mellon University, (1988).
Reiner, M., Chi, M. and Resnick, L. Naive Materialistic Belief: An Underlying Epistemological Committment. *Proceedings of the Cognitive Science Society Conference 1988*, 544 (1988).
Smith, C., Carey, S. and Wiser, M. On Differentiation: A Case Study of the Development of the Concepts of Size, Weight and Density. *Cognition* **21**, 177(1985).
Thornton, R. Tools for Scientific Thinking: Microcomputer-Based Laboratory, *Physics Education* (1987).
Viennot, L. Spontaneous Reasoning in Elementary Dynamics. *European Journal of Science Education* **1**, 205(1979)

Acknowledgements

Work supported by NSF Grants MDR-8954733 and MDR-8950313.

The Effect of Feedback Control on Learning to Program with the Lisp Tutor

Albert T. Corbett
John R. Anderson

Department of Psychology
Carnegie-Mellon University

Abstract

Control and content of feedback was manipulated as students practiced coding functions with the Lisp Tutor. Four feedback conditions were employed: (1) immediate error feedback and correction, (2) immediate error flagging but immediate correction not required (3) feedback on demand and (4) no tutorial assistance. The wide range in feedback conditions did not affect mean learning rate as measured by individual production firing, time to complete the exercises or post-test performance. However, post-test results were more highly correlated with student ability as tutorial assistance decreased across conditions. Feedback conditions also affected students' monitoring of the learning process. Across groups, students found the material was easier and believed they had learned it better as assistance decreased across conditions. However, students who received more assistance estimated their mastery of the material more accurately. Finally, students reported relatively little preference for one tutoring condition over the others. Students who could exercise the most control over feedback reacted fairly passively to the tutoring conditions; students in condition 3 tended not to ask for much help and students in condition 2 tended to correct error immediately although it was not required.

Introduction

This study examines four types of feedback in the context of students learning to program with the Lisp Tutor. The four conditions vary from fairly rigid step-by-step feedback to a condition in which no feedback is available. We are interested in several theoretical and practical issues in this research: (1) whether immediate feedback is optimal for learning as predicted by an earlier version of ACT* (2) whether feedback conditions affect students monitoring of how well they have mastered the material and (3) whether students have preferences among the feedback conditions.

The Lisp Tutor is a program that provides assistance to students as they work on lisp coding exercises (Anderson & Reiser, 1985). As the students type their code, the tutor monitors their progress step-by-step, provides feedback on errors and provides the correct next action when a student appears to be floundering. The tutor contains a production system ideal student model and provides feedback by comparing the students responses to correct and buggy productions that could fire at each step. The tutor was developed in large part to test the ACT* model of procedural learning in a "real-life" context and has also been used to explore tutorial variables (Anderson, Conrad & Corbett, 1989)

The standard lisp tutor constrains students to enter their code top-down, depth-first and left-to-right, and presents immediate, atom-by-atom feedback on errors. When an error is made, the tutor interrupts, presents a feedback message, deletes the error and requires the student to try again. An early version of the ACT* model suggested that immediate feedback should facilitate the proceduralization of declarative knowledge, since the formation of productions depends on the working memory trace of the problem solving episode (Anderson, Farrell & Sauers, 1984; Anderson, Boyle, Farrell & Reiser, 1987). However, immediate feedback does not uniformly enhance learning (Kulik & Kulik, 1988) and in earlier studies with the Lisp Tutor, removing feedback content or delaying feedback has not had any impact on degree of learning as assessed in a post-test (Anderson, Conrad & Corbett, 1989; Corbett & Anderson, 1989; Corbett, Anderson & Patterson, in press). Moreover, in some circumstances, immediate feedback can interfere with the students' monitoring of the learning process (Schmidt, Young, Swinnen & Shapiro, 1989). As a result, in this study we have implemented four feedback conditions. Unlike earlier studies, these conditions are implemented in the same program architecture, making it possible to compare production firing times. In addition to this direct test of the theory, we measure post-test performance, time to complete the exercises and have collected questionnaire data.

The four feedback conditions are (1) standard immediate feedback, as described above; (2) error flagging, (3) feedback on demand and (4) no feedback (see Corbett, Anderson & Patterson for implementation details). The three non-standard conditions differ from the standard immediate feedback condition in two ways. First, in these three conditions, students entered their code with a true structured editor. Thus, they could type their code in any order, and edit their code at will. Second, the conditions varied with respect to feedback content and control

In the error flagging condition, the tutor provided immediate feedback, but did not interrupt the students. It flagged any code it did not recognize by displaying it in boldface, but did not display any explanatory text and the students remained in control of their actions. They were free go back and fix the error, ask for a comment on the error (the same message provided automatically in the immediate feedback condition) or to continue coding. The tutor recognizes one or more roughly optimal solutions to the exercises. Errors are flagged on the basis of this knowledge, so, in effect, the tutor is guiding students to an optimal answer. However, all three non-standard versions ultimately will accept any code that works and students knew in this condition that they might have a workable solution, even if some of the code were flagged. If the students completed an exercise with working, but unrecognizable code, the tutor also displayed one of its optimal solutions. There are possible theoretical and practical advantages in this condition. First, students may benefit from generating their own explanations of errors, instead of automatically receiving feedback. Second, in the course of learning, larger productions may be compiled. In that case, atom-by-atom feedback could interrupt the firing of productions. If so, this version of the tutor allows students to complete a production before pausing to deal with an error.

In the feedback on demand condition, the tutor never interrupted the student. However, the student could ask the tutor at any time to check over the code. In this case, the tutor checked the code top-down and as soon as it found an error, provided the same feedback message that the standard tutor would have presented automatically when the error is made. If there were no errors, it informed the subject accordingly. (It should be noted that in all three versions of the tutor that have been discussed so far, the student could also ask for two other types of help: (1) a hint at any goal in the solution (2) the correct action to take at any goal.) If students benefit from detecting as well as correcting their errors, this feedback on demand condition could prove to be optimal.

In the no feedback condition, no tutorial assistance was available to the students as they generated their code. Students in all conditions had access to a lisp environment during coding, so they could load and test their code, but this was the only mechanism available to students in this condition for debugging. In the three non-standard conditions, the students indicated their solutions were complete by pressing a key. In the error flagging condition and feedback on demand conditions, students could not move on to the next exercise until their solution was correct. Students in the no tutor condition were ultimately allowed to move on with incorrect code, but only after trying at least five times to get the correct answer. If the student gave up, the tutor showed them a correct solution.

The Experiment

This experiment was conducted in a class and a total of 55 students participated. Each student worked with one version of the tutor. The four versions of the tutor were implemented for the first two lessons of the lisp tutor. These lessons cover an introduction to function calls and to function definitions, and consist of a total of 20 exercises. While some of these exercises are difficult, most students are ultimately capable of generating solutions even without help from the tutor. After completing these lessons, all students reverted to the standard tutor and so had the same learning environment for the remaining 10 lessons in the course. A paper and pencil post-test was administered after the first two lessons. Four more paper and pencil tests were administered throughout the rest of the course. Since all students had the same learning experiences over the last ten lessons these last four tests will serve as a measure of relative ability. In addition, questionnaires on the tutor were administered after the second and fourth lessons.

Results

Learning. Scores on the post-test following lesson 2 are displayed in Table 1. The effect of tutor type is not significant in an analysis of variance. Because of attrition and absence, the students in the four groups were not well matched. As a result, we used students' mean performance on the remaining tests in the course as a measure of relative ability and removed the effects of this measure in an analysis of covariance. The adjusted means are also displayed in Table 1 and again the effect of tutor type is not significant. Average time to complete each exercise is also displayed in Table 1. Again, there are no significant differences in an analysis of variance or an analysis of covariance that removes the effects of students' relative ability.

Table 1

Mean Post-Test Scores (percent correct) and Mean Exercise Completion Times (minutes) for the Four Versions of the Tutor.

	Immediate Feedback	Error Flagging	Demand Feedback	No Tutor
Post-Test Scores	55%	75%	75%	70%
Adjusted Scores (ANCOVA)	61%	75%	71%	67%
Exercise Times	4.6	3.9	4.5	4.5
djusted Times (ANCOVA)	4.1	4.2	4.5	5.0

There is no evidence in these global measures that the feedback manipulation affected learning. To analyze the effect of feedback on learning more closely we also examined production firing accuracy and firing time in the course of coding. Accuracy is operationally defined as the probability that the student types an atom that corresponds to an appropriate underlying production in the tutor's ideal student model. Production firing time is only measured for correct coding cycles. It is operationally defined as the time from the onset of the prompt at the beginning of a coding cycle until the onset of the prompt for the next cycle. Table 2 displays mean time and accuracy data for the first through fifth opportunities to fire each production. As can be seen accuracy increases and firing time decreases with repeated production firings. Note that there is a speed accuracy tradeoff between the

Table 2

Production Firing Accuracy (percent correct) and Production Firing Time (seconds) in Lesson 2.

	First Occurrence	Second Occurrence	Third Occurrence	Fourth Occurrence	Fifth Occurrence
Immediate Feedback					
Accuracy	83%	90%	95%	95%	91%
Time	18.9	14.5	8.9	8.2	8.9
Error Flagging					
Accuracy	62%	69%	82%	73%	81%
Time	11.5	6.0	5.4	4.1	3.0
Feedback on Demand					
Accuracy	59%	52%	88%	82%	70%
Time	15.7	9.0	6.7	4.9	4.9
No Tutor					
Accuracy	59%	67%	80%	78%	76%
Time	15.0	9.2	7.4	6.7	4.6

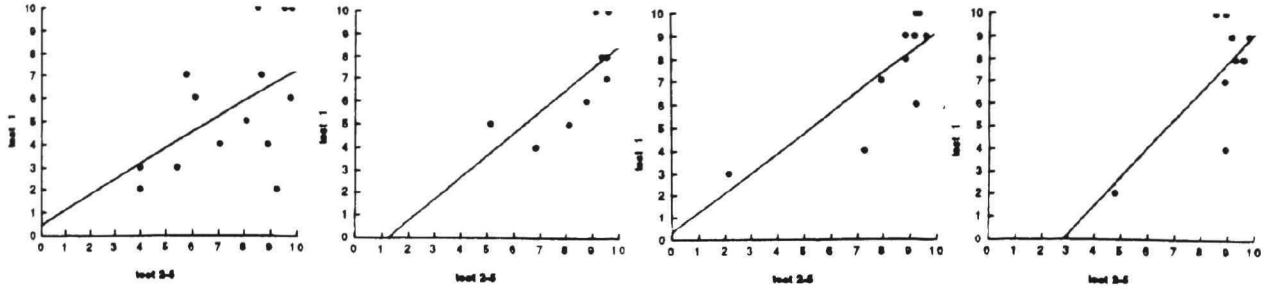
three non-standard conditions and the immediate feedback condition. Students in the immediate feedback condition are responding more accurately at each serial position, but they are also responding more slowly. These students are more cautious in the immediate feedback condition, perhaps because the tutor reacts strongly to each mistake in this condition, but there is no evidence that they are learning the productions better. Thus, it seems that the process of

Conrad & Corbett (1989) and in contrast to the earlier proposals of Anderson, Boyle, Farrell & Reiser (1987). The speed accuracy tradeoff itself is interesting, since there is currently no mechanism in ACT* that allows such a tradeoff in firing productions.

Finally, to assess whether ability interacted with tutoring condition, we we plotted Test 1 accuracy against our ability measure (students' performance on the remaining tests in the course) in Figure 1. The slopes generally grow progressively steeper and the goodness of fit generally increases as the the control exerted by the tutor diminishes. The standard immediate feedback condition has the lowest slope, while the slope for the no tutor condition is almost twice as large. The two other versions fall in between. This result suggests, as might be expected, that when less assistance is provided, student's performance depends more heavily on relative ability.

Figure 1

Test 1 Accuracy Plotted Against Average Accuracy on Remaining Tests



Immediate Feedback	**Error Flagging**	**Feedback on Demand**	**No Tutor**
Slope=0.68	Slope=0.97	Slope=0.88	Slope=1.27
r=0.50, p<.10	r=0.70, p<.05	r=0.80, p<.01	r=0.72, p<.01

Questionnaire 1: Monitoring the Learning Process. A questionnaire was administered after students completed the first two lessons with the tutor, but before taking the post-test. The questionnaire contained seven questions to which students responded on a seven point scale (See Table 3). Two of the questions concerned the students perception of the learning process, since to the ability to self-monitor this process is an important skill in students (Chi, Bassok, Lewis, Reimann & Glaser, 1989). These are the first two questions in Table 3. The first question asked how difficult the exercises were and the differences between the groups were marginally significant ($F(3,51) = 2.25$, $p < .10$) As can be seen in Table 3, there is a perfect correlation between degree of tutorial assistance and ratings of difficulty. (Pairwise comparisons between the no feedback and error flagging, and between no feedback and standard tutor were reliable). The paradoxical result is that the more assistance provided by the tutor, the harder the exercises seemed. It appears that perceived difficulty reflects not the degree of effort expended by the student, but rather the degree to which an external source points out the students' errors. We generated scatter plots for this question to examine the relationship of the ratings to student ability. In Figure 2, the seven point rating scale appears on the abscissa and student performance on the post-test appears on the ordinate. As can be seen, the slope of the best fitting straight line is steeper for the standard and error flagging condition, both of which provide immediate feedback, than for the feedback on demand and no tutor conditions. While the immediate feedback groups perceived the questions as more difficult, it appears that the students within the two immediate feedback conditions were more sensitive to the difficulties they experienced. Moreover, in the feedback on demand and no tutor groups the correlation between difficulty estimates and post-test performance is small and nonsignificant.

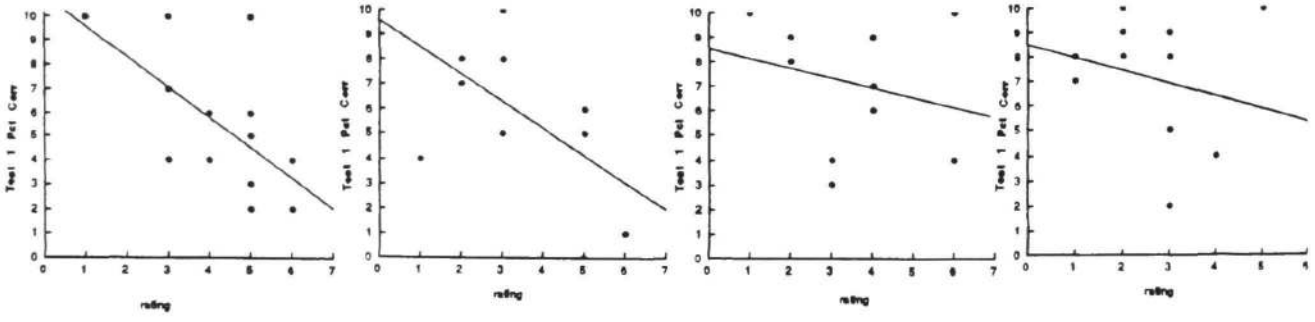
Table 3

Questionnaire 1 Mean Ratings

	Imm Fdbk	Error Flag	Demand Fdbk	No Tutor
1. How difficult were the exercises? (1=Easy, 7=Challenging)	4.1	3.9	3.4	2.8
2. How well did you learn the material? (1=Not Well, 7=Very Well)	5.4	4.6	5.4	5.8
3. How much did you like the tutor? (1=Disliked, 7=Liked)	5.2	4.5	4.8	4.9
4. Did the tutor help you finish? more quickly (1=Slower, 7=Faster)	5.1	4.6	4.7	4.5
5. Did the tutor help you understand better? (1=Interferred, 7=Helped)	5.3	4.9	4.7	4.7
6. Did you like the tutor's assistance? (1=Disliked, 7=Liked)	5.3	5.0	4.7	4.7
7. Would you like more or less assistance? (1=Less, 7=More)	4.3	4.9	4.5	4.6

Figure 2

Test 1 Accuracy Plotted Against Ratings of the Difficulty of the Lessons



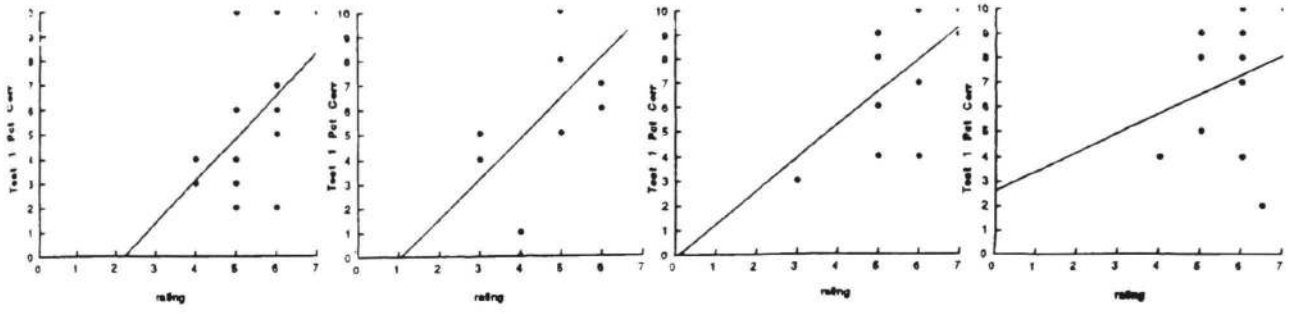
Immediate Feedback	**Error Flagging**	**Feedback on Demand**	**No Tutor**
Slope=-1.27	Slope=-1.10	Slope=-0.40	Slope=0.52
r=0.62, p<.05	r=0.53, p<.10	r=0.26, ns.	r=0.24, ns.

The second question asked how well the student had learned the material. Again, there was a significant effect of tutor type, ($F(3,51)=3.57$, $p < .05$). (Pairwise comparisons between the error flagging and no tutor conditions and between the error flagging and standard condition were significant). The results of this question are similar to what would be predicted from the prior question; generally the groups that thought the material was easy also thought they had learned it better. The exception is that the standard immediate feedback condition has shifted upward.

This may reflect the fact that this is the only group that received a written explanation from the tutor each time they made a mistake. Scatter plots for this question are displayed in Figure 3. Again, the slope is higher for the two groups receiving immediate feedback, suggesting, that the subjects in those groups are more sensitive to how well they'd learned the material.

Figure 3

Test 1 Accuracy Plotted Against
Ratings of How Well the Material was Learned.



Immediate Feedback	**Error Flagging**	**Feedback on Demand**	**No Tutor**
Slope=1.75	Slope=1.64	Slope=1.34	Slope=0.79
r=0.52, p<.05	r=0.51, p<.10	r=0.59, p<.10	r=0.24, ns.

Questionnaire 1: Tutoring Preferences. The remaining five questions concerned the assistance provided by the tutor. The surprising result is that despite the wide range of control exerted by the tutor across the four conditions, students did not display any preferences among the tutoring conditions. There were no significant differences among the four groups in rating (1) how much they liked working with the tutor, (2) how much help the tutor was in completing the exercises, (3) how well they liked the tutor's assistance and (4) whether they would prefer more or less assistance.

An interesting aspect of this issue is that the students in the two groups who were able to exercise control, the error flag and feedback on demand groups, gravitated toward the two extremes. In the case of the feedback on demand tutor, only 3 of 11 subjects ever asked the tutor to check over code (only 2 asked more than once), 2 asked for goal hints and 4 asked for any explanations. In the case of the error flagging condition, subjects on average corrected 58% of their errors immediately and 76% after no more than one intervening operation. Thus, when given control, students respond fairly passively. If the tutor does not volunteer help, the students tend not to ask for it. If the tutor flags errors, students tend to respond immediately, even though they are free to do otherwise.

Questionnaire 2. After students had all worked with the standard immediate feedback tutor for two lessons, we asked them to fill out a second questionnaire comparing the two tutors they had used. This questionnaire contained four questions, displayed in Table 4, that required a rating on a seven point scale. As can be seen in question 1, all four groups preferred the first version of the tutor. This may seem anomalous for the group that used the standard tutor in the first place. However, beginning in the third lesson, students no longer had access to a lisp environment while doing exercises, and as can be seen in question 4, all four groups liked having the Lisp environment available. The second question asked about the editing facilities and there was a marginally significant effect of tutor type here, $F(3,48)=2.19$, $p=.10$. In general, students who had a true structured editor preferred that interface to the more constrained interface in the standard condition. Finally, the third question asked about the feedback and help properties of the tutor and, surprisingly all four groups rated themselves as indifferent between the tutor they used for the first two lessons and the standard immediate feedback tutor used subsequently.

Table 4

Questionnaire 2 Mean Ratings

	Imm Fdbk	Error Flag	Demand Fdbk	No Tutor
1. Which tutor do you like better? (1=Version 1, 7=Standard Tutor)	2.9	2.0	2.6	2.7
2. Which editing facilities did you prefer? (1=Version 1, 7=Standard Tutor)	3.4	2.0	3.0	2.3
3. Which feedback and help facilities did you prefer? (1=Version 1, 7=Standard Tutor)	4.1	3.6	3.1	3.9
4. Did you like having the Lisp environment? (1=Yes, 7=No)	2.5	2.2	2.1	2.1

In summary, six questions on tutorial assistance did not reveal any preferences. There are two points that qualify this conclusion. First, when students are given a structured editor to enter their code, they almost always conform to the top-down, left-to-right, depth-first constraint imposed by the tutor. They only deviate to go back and fix errors. This suggests that the preference for the true structured editor expressed in question 2 actually reflects a preference for detecting and fixing their own errors. Second, an earlier study (Corbett & Anderson, 1989) compared just the standard tutor to the error flagging tutor and asked specifically how much students liked the *immediate* feedback presented by the tutor. In that study, students in the error flagging condition gave a more positive rating than those in the standard condition. Thus, there is some evidence that students do not like being interrupted when they make mistakes.

Conclusions

In summary a wide range of tutor vs. student controlled feedback led to no differences in how quickly nor how well students learned how to write Lisp code. This is consistent with our current view (Anderson, Conrad & Corbett, 1989) that learning is a function of the product of a problem solving episode and not the process. As long as the students produce the same code and come to the same understanding of the code, it does not matter what cognitive trajectory they took to produce the code. As our tutorial manipulation is extended to more complex functions, we expect that the feedback manipulation will begin to affect time to complete the exercises. However, we expect that production formation and post-test performance will not be affected.

While the manipulation of feedback seemed not to affect learning, it did influence the students self-perception. Students who received less assistance from the tutor seemed more confident of the skill, however that confidence tended to be unrelated to performance on the post-test. Students who received immediate feedback seemed to assess their knowledge more realistically.

Concerning students' attitudes, there is a general preference for an enhanced feeling of control, even though students don't take much advantage of the added flexibility. When given a structured editor they still tend to type their code top-down, left-to-right and depth-first. When provided a Lisp environment less than half the students use it, except in the no tutor condition. Finally, students apparently prefer to find and fix their own errors and if errors are pointed out, they prefer not being interrupted. However, none of the feedback manipulations affected how well the material was learned.

References

Anderson, J.R., Boyle, C.F., Farell, R. and Reiser, B.J. (1987). Cognitive principles in the design of computer tutors. In P. Morris (Ed.) Modelling Cognition (pp. 93-134). New York: Wiley.

Anderson, J.R., Conrad, F. and Corbett, A.T. (1989). Skill acquisition and the LISP Tutor. Cognitive Science, 13, 467-505.

Anderson, J.R., Farrell, R. and Sauers, R. (1984). Learning to program in LISP. Cognitive Science, 8, 87-129.

Anderson, J.R., and Reiser, B.J. (1985). The LISP Tutor. Byte, 10, 159-175.

Chi, M.T.H., Bassok, M., Lewis, M.W., Reimann, P., Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. Cognitive Science, 13, 145-182.

Corbett, A.T. and Anderson, J.R. (1989). Feedback timing and student control in the Lisp Intelligent Tutoring System. The Proceedings of the Fourth International Conference on AI and Education., Amsterdam, Netherlands.

Corbett, A.T., Anderson, J.R. and Patterson, E.G. (in press). Student modeling and tutoring flexibility in the Lisp Intelligent Tutoring System. In C. Frasson and G Gauthier (eds.) Intelligent tutoring systems: At the crossroads of artificial intelligence and education. Norwood, NJ: Ablex.

Kulik, J.A. & Kulik, C.C. (1988). Timing of feedback and verbal learning. Review of Educational Research, 58, 79-97.

Schmidt, R.A., Young, D.E., Swinnen, S., & Shapiro, D.C. (1989). Summary knowledge of results for skill acquisition: Support for the guidance hypothesis. Journal of Experimental Psychology: Learning, Memory and Cognition, 15, 352-359.

Supporting Linguistic Consistency and Idiosyncracy with an Adaptive Interface Design

Jill Fain Lehman
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
jef@f.cs.cmu.edu

Abstract

Despite the goal to permit freedom of expression, natural language interfaces remain unable to recognize the full range of language that occurs in spontaneously generated user input. Simply increasing the linguistic coverage of a large, static interface is a poor solution; as coverage increases, response time decreases, regardless of whether the extensions benefit any particular user. Instead, we propose that an *adaptive interface* be dedicated to each user. By automatically acquiring the idiosyncratic language of each individual, an adaptive interface permits greater freedom of expression while slowing system response only insofar as there is ambiguity in the individual's language. The usefulness of adaptation relies on the presence of three regularities in users' linguistic behaviors: within-user consistency, across-user variability, and limited user adaptability. We show that these behaviors are characteristic of users under conditions of frequent use.

1. Introduction

Research in designing natural language interfaces attempts to facilitate human-computer interaction by allowing a user the power and ease of her usual forms of expression to accomplish a task. Unfortunately, current technology falls short of this ideal. No existing interface permits all of the complexities of natural language: including the full range of potentially relevant vocabulary, idiomatic phrases, and extra-grammatical constructions. In addition, a tension exists between providing full freedom of expression and providing reasonable response time. Due to language's local ambiguity, significant increases in the coverage of a grammar are usually accompanied by intolerable decreases in system performance.

The standard resolution to this tension is the "monolithic" interface. For a given domain and task, the designer must choose a restricted sublanguage that has the property of being both computationally tractable and "user-friendly" for all users. Three assumptions underlie such a design:

- It is desirable to build a single interface for all users.
- There is a tractable subset of language for the domain and task that is both common to and natural for all users.
- If a user's preferred form of expression differs from the common subset, the user can easily adapt her style to conform to the limitations imposed by the interface.

How well does an interface based on these assumptions serve the individual? To the extent that a user's preferred forms of expression lie outside the restricted subset, the system relies on her adaptability and is, therefore, no more facilitating of the interaction than a system-supplied command language. In other words, this design sacrifices expressibility for responsiveness. On the other hand, to the extent a user's preferred forms of expression lie inside the restricted subset but correspond to only a small fraction of the sublanguage, the user pays a price in response time for the system's ability to understand utterances she will never use. Thus the design also sacrifices responsiveness for expressibility.

As an alternative, we propose the *adaptive interface* design. Under this paradigm each user has a dedicated interface capable of understanding her idiosyncratic language and limited in its responsiveness only by the ambiguity inherent in that language. This single-user approach is possible through a technique called *adaptive parsing*, a learning method in which the system automatically acquires a user's preferred forms of expression by growing its kernel grammar dynamically in response to user interactions. In essence, the learning mechanism explains perceived errors in the input utterance as one or more deviations with respect to the system's current grammar (a deviation corresponds to an insertion, deletion, substitution, or transposition of text with respect to a known grammatical form). The explanation is then transformed into new grammatical components that are capable of recognizing the general structure of the deviation in future interactions.

The practicality of this approach has been demonstrated by the implementation of an adaptive interface called CHAMP which acquired eight very different idiosyncratic grammars from users' spontaneously generated input (see [4, 5]). The assumptions upon which CHAMP is based differ significantly from those of a monolithic design. In the remainder of this paper we examine the assumptions underlying adaptive parsing and demonstrate their validity.

2. Behavioral Hypotheses

There are three conditions necessary for an adaptive interface to benefit user performance. Each condition corresponds, in turn, to an hypothesis about users' linguistic behavior:

- **Condition 1: within-user consistency.** We hypothesize that in a natural language interface used frequently by the same individual over time, the user tends to rely on those forms of expression that she remembers as having worked in the past. In other words, frequent use leads to a *stylized* and *self-bounded* grammar. If the hypothesis is false and a user does not rely primarily on what worked in the past, then a fixed, restricted sublanguage is necessary—a system based on automatic adaptation would have no protection against the intractability associated with steadily increasing coverage.
- **Condition 2: across-user variability.** We hypothesize that given an environment that permits a reasonable subset of English, users will demonstrate significant idiosyncracy in their preferred forms of expression. If this hypothesis is false and all users employ the same restricted sublanguage then a methodology such as that of Kelley [3] or Good *et al.* [1] is preferable; one builds a monolithic system incorporating the complete common subset through a generate-and-test cycle.
- **Condition 3: limited user adaptability.** A monolithic approach to interface design assumes that the cognitive burden of learning the interface's sublanguage is a minor one. We hypothesize that a system that permits the natural expression of idiosyncratic language patterns results in better task performance than a system that forces the user to adapt to it. If our hypothesis is wrong and the user is able to adapt quickly and with little effort to a restricted sublanguage, then adaptation on the part of the system is no longer a clear advantage.

The experiments discussed in the next section validate our three behavioral hypotheses and prove that with frequent use the required conditions are met.

3. Description of the Experiments

Data was collected at two distinct points in our research on adaptive interfaces. To demonstrate the feasibility of adaptive parsing and to establish the validity of the behavioral hypotheses, we first simulated an adaptive interface using a hidden-operator paradigm.[1] The algorithm used by the hidden operator formed the basis for the implementation of the working interface, CHAMP. Next, to verify that the implementation captured the crucial aspects of the simulated environment (and to assure ourselves that our initial results did not depend on having a human in the loop), we used CHAMP to run the initial experiments "on-line" with two new users.

Materials. A calendar scheduling task was chosen because it has fairly well-defined semantics and requires frequent interactions over time. The stimuli consisted of calendar pages with pictorial representations of changes to be made to the schedule. The stimuli was pictorial in order to minimize its influence on the user's forms of expression.

Users. All of the users were female adults between twenty and sixty-five years of age. None had prior experience with natural language interfaces. Each was employed as a secretary or executive assistant in either a business or university environment. Although not every user maintained a calendar for her employer, each had kept a personal calendar for at least one year.

Procedure. The *Adapt* condition was designed to test our hypotheses about the development of self-bounded, idiosyncratic grammars. Users were told that they would be typing the input to a *natural language learning*

[1] A hidden-operator experiment is one in which the user believes she is interacting with a computer system when, in reality, the feedback is produced by an experimenter simulating the system's behavior. For examples of other studies conducted under the hidden-operator paradigm, see [1], [3], [9], and [10]. The hidden-operator experiments described here were first reported in [6].

interface that would increase its knowledge of English while helping them keep an on-line calendar for a busy professor/entrepreneur. In each session, the user was asked to effect the changes in ten to twelve task pictures by typing her commands as if she were "speaking or writing to another person." Although no time limit was enforced, the instructions told users to proceed to the next task after three unsuccessful attempts.

In responding to a user's utterances, the hidden operator or CHAMP had the user's current grammar and lexicon available. With respect to that information, an utterance was judged either *parsable*, *learnable* (containing at most two deviations), or *uninterpretable* (more than two deviations). A parsable utterance was accepted without further interaction and the desired action was performed. The interpretation for a learnable utterance had to be verified by the user. If verified, the current grammar was augmented to capture the new forms, and then the action was performed. If the utterance was uninterpretable, a user in the hidden-operator experiments was told which segments, if any, appeared understandable and was asked to try again. In the on-line experiments, users were simply asked to try again. Each of the six users in the Adapt condition participated in nine sessions; Users 1, 2, 3, and 4 took part in the original hidden-operator experiments, Users 9 and 10 participated in the later on-line experiments.

To evaluate the counterargument to adaptation that maintains that the user will naturally adapt to the system's limitations faster than the system can usefully adapt to the user's idiosyncrasies, two users participated in the *No-Adapt* control condition. In this condition, the experiment was conducted as outlined above except that users were told the system was a natural language interface (not a learning interface), and the kernel grammar was never changed. Although the system remained more permissive of extragrammaticality than the average natural language interface (by allowing up to two ungrammatical constructions with respect to the kernel grammar), the user was nevertheless restricted to a fixed sublanguage. Any improvement in task performance would therefore be attributable to the user's adaptability. The number of sessions varied in the control condition according to the availability of the users. User 6 participated in three sessions, User 8 in five sessions.[2]

4. Results and Discussion

With the exception noted below, all users were able to complete most tasks; they rarely required more than three attempts per task in the initial sessions and averaged only slightly more than one attempt per task in later sessions. The results for users in the Adapt condition indicated a high degree of self-limiting behavior, converging towards a user-specific subset of English. Users in the No-Adapt condition showed very limited adaptability. In the remainder of this section we review the results pertaining to each behavioral hypothesis in turn.

Within-user consistency. To measure this aspect of user behavior, we examine the number of constituents added to an individual's grammar over time. If a user is self-bounding in her language, then this number should decrease asymptotically. Figure 4-1 shows the measure for each of the six users in the Adapt condition. The figure indicates, by user and session, the number of new constructions per *interpretable* sentence which is calculated as the total number of changes to the grammar divided by the total number of *parsable* or *learnable* sentences. The number of *uninterpretable* sentences in a session is given in parentheses. Shaded areas of the figures indicate sessions following the introduction of the supplementary instructions to work quickly.[3]

---

[2]Users 5 and 7 participated in a different control condition (*Adapt/Echo*) which was run in response to arguments by Slator *et al.* [9] that users want to and will learn a mnemonic command syntax and domain-specific vocabulary. Users in the Adapt/Echo condition were given the same instructions as those in the Adapt condition except that they were told the system would display a paraphrase of their utterance in an "internal form" that they were free to incorporate into their commands or to ignore, as they saw fit. No experimental effect was observed in this condition. Further discussion can be found in [4].

[3]The supplementary instructions were included in response to the observation that users try to employ terser, more economical language as time goes on [7, 2, 3, 8]. It was unclear whether nine sessions would be adequate time for the tendency to be manifested. Thus, the additional instructions were included to subject the system to more extreme linguistic behavior within the time available. Note, however, that the supplementary instructions were not given until the user's grammar had essentially stabilized. Figure 4-2 shows examples of movement toward terser language by our users.

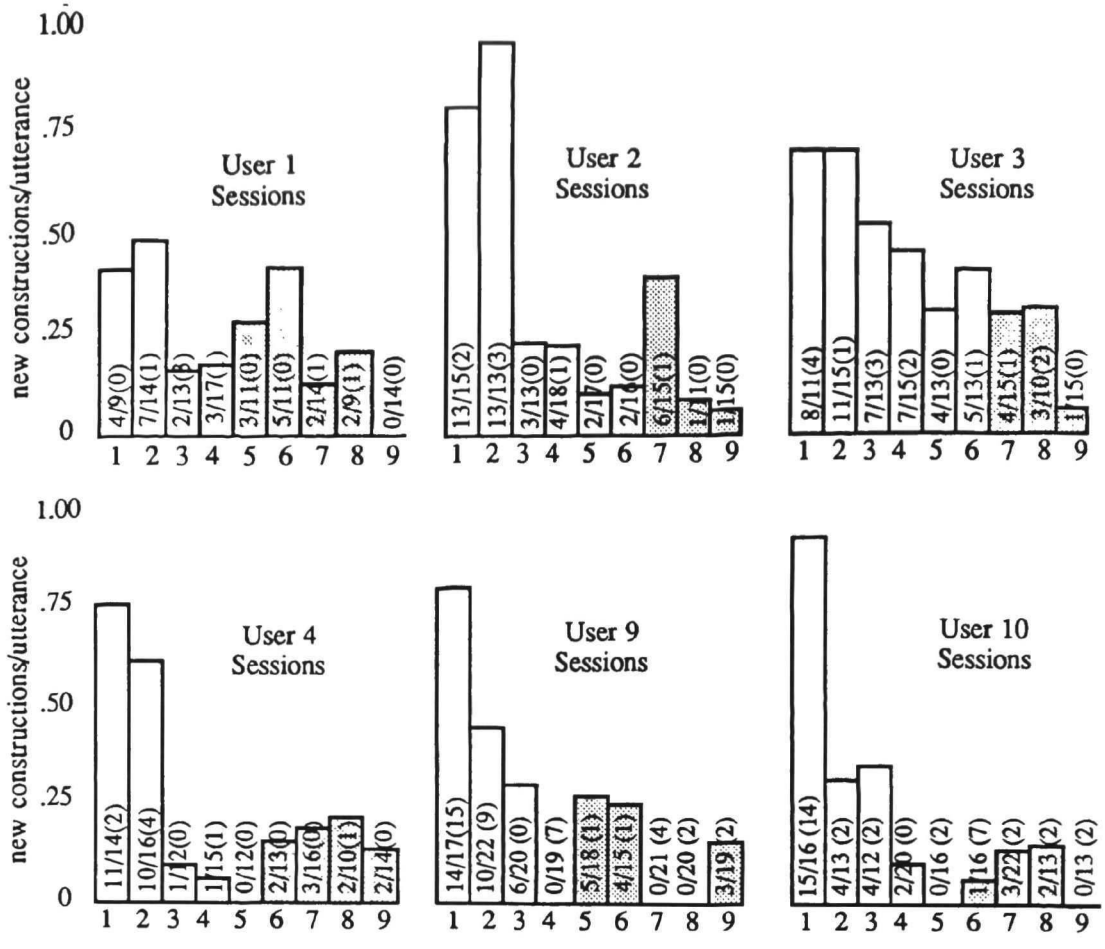


Figure 4-1: Self-limiting behavior in the Adapt condition. Shaded sessions followed the supplementary instructions to work quickly. The number of unparsable utterances for each session is given in parentheses.

Figure 4-1 shows both the effect of the choice of kernel forms and the difference in the rates at which language stabilization occurs. Observe that User 1's grammar required the least learning and stabilized quickly. She began with terse utterances and a vocabulary that was quite close to the kernel chosen for the experiment. User 2's grammar showed less initial resemblance, but she employed it consistently and, thus, stabilized quickly as well. User 3 had a tendency to be both polite and verbose, so her utterances were more likely to contain new forms and stabilization occurred more slowly. User 4 showed rapid stabilization because the new forms she introduced in sessions one and two, while not terse, were quite "natural" (that is, they were used consistently and without modification in subsequent sessions).

Users 9 and 10 showed the same overall decrease in the number of new constructions per interpretable sentence as the users in the hidden-operator experiments. They also showed the same local increase and return to stabilization after the supplementary instructions were given. User 9's behavior was particularly interesting. Despite the fact that the instructions clearly stated that the system could not understand its own output, she insisted on using the system's messages as a template for her own commands (User 7 showed a less extreme form of the same behavior in the hidden-operator experiments). By the end of her third session, User 9 had essentially used the adaptive qualities of the interface to bootstrap the system to understand a terser form of its own output.

Across-user variability. The high degree of idiosyncracy in users' preferred forms of expression can be seen in Figure 4-2. Note that much of that idiosyncracy is maintained despite the movement over time to terser utterances. For users in the Adapt condition of the hidden-operator experiments, the grammars resulting from

interactions with the simulated interface show little overlap. One could argue, however, that the idiosyncracy found is an artifact of the hidden operator's freedom when adding a new form to a user's grammar. CHAMP's uniform adaptation and generalization mechanism provide the opportunity to examine across-user variability in a more meaningful way.

u1.1.9	move Anderson seminar on June 10 to room 7220
u1.9.3	move 1:30 PM class on June 23 to room 5409
u2.1.14	change the location of the meeting on June 10, 1986 at 12:00 noon ending at 1:30 from room 5409 to 7220
u2.9.3	change Class 15-731 location to Room 5409 on June 23
u3.1.13	CogSci Seminar will be held on Tuesday, June 10, 1986 from 12:30 to 1:30 in room 7220; Speaker will be Anderson.
u3.9.3	Class 15-731 will be held in Room 5409 instead of 7220 on June 23rd.
u4.1.13	change room number of seminar from 5409 to 7220 on Tuesday June 10 from 12:30 to 1:30
u4.9.3	class 15-731 on Monday, June 23 will be in room 5409
u9.1.27	change location of COGSCI Seminar to Room 7220 June 6
u9.9.5	change class 15-731 June 19 from 7220 to 5409 at 1:30pm
u10.1.26	Change June 6 Cogsci seminar to WeH 7220
u10.9.3	Change June 19 class to room 5409

Figure 4-2: Sample sentences from sessions one and nine for users in the Adapt condition (u*i.j.k* refers to the *k*th utterance in session *j* for user *i*).

To attain a quantitative measure for idiosyncracy, we examine the acceptance rate for each user's total set of utterances across the nine sessions using her own final grammar (her grammar at the end of session nine) and using each of the other users' final grammars. If each final grammar recognizes approximately the same language, then there should be no significant difference in the acceptance rates. In contrast, Figure 4-3 clearly demonstrates a wide range of variability. For example, User 2's final grammar is able to parse 88% of her own sentences, but can parse only 11% of the utterances from User 3.[4] Note that in no instance does the grammar for another user approach the acceptance rate of the user's own final grammar. The high values on the diagonal, which denote the acceptance rate of each user's final grammar for her own utterances, result both from within-user consistency and from the efficacy of the learning mechanism.

CHAMP also allows us to quantify the degree to which idiosyncracy contributes to poor response in a monolithic interface design. Recall that in motivating the idea of adaptive parsing, we argued from a theoretical point of view that the monolithic approach must engender some degree of mismatch between the language accepted by the system and the language employed by any particular user. The mismatch is of two types: language the user prefers that cannot be understood by the interface and language the interface understands that is never employed by the user. To compare designs, we must control for the first kind of mismatch by guaranteeing that the monolithic grammar can parse at least as many sentences in the test set as can CHAMP. To accomplish this, we construct a monolithic grammar by adding to the kernel the union of the users' final grammars minus any redundant components.

Figure 4-4 demonstrates the monolithic grammar's performance on each user's utterances. The figure provides values for three measures: the average number of search states, the average number of roots (each root

[4]We do not compare final grammars across experiments because the on-line users started with a slightly different kernel grammar than the hidden-operator users; a cross-experiment comparison would show inflated variability due to those differences.

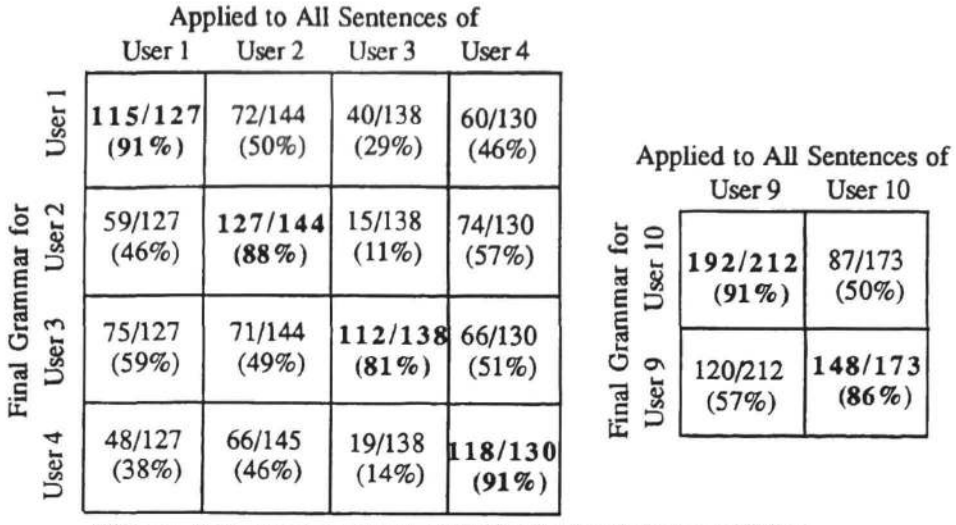


Figure 4-3: Across-user variability in the Adapt condition.

represents a unique parse tree; multiple roots correspond to multiple interpretations), and the number of utterances parsable by the monolithic grammar that were not also parsable by the user's final grammar. Due to the implementation, increased values for the first two measures (states and roots) can correspond to relatively independent sources of ambiguity in the grammar. From the user's point of view, an increase in the number of states corresponds to decreased response time, while a higher number of roots corresponds to an increase in the number of interactions required to arrive at a unique interpretation for the utterance.

User	Avg. states$_{user}$	Avg. states$_{mono}$	Avg. roots$_{user}$	Avg. roots$_{mono}$	Additional$_{mono}$
1	37.6	109.6	1.5	7.7	4
2	59.8	155.5	1.6	8.5	4
3	43.3	156.5	2.0	9.3	1
4	40.7	148.3	1.5	7.1	1
9	80.6	96.7	2.6	2.6	0
10	45.8	68.2	1.7	3.4	0

Figure 4-4: A comparison of the relative costs of parsing with a monolithic grammar versus parsing with the idiosyncratic grammars provided by adaptation.

Let us first consider the results for the users in the hidden-operator experiments. Figure 4-4 shows that in every instance the average number of search states examined by the monolithic grammar is significantly greater than the number required by the user's idiosyncratic grammar. Although we do gain the ability to understand ten additional sentences in the test set (this occurs when adaptations learned for User *i* are able to parse a sentence that was rejected for User *j*), the trade-off hardly seems favorable. User 1 would find a 3% increase in the number of her sentences understood by the interface (4/127) at a cost of almost three times the search. User 2 also receives about a 3% increase (4/144) at a cost of about two and a half times the search. User 3 and User 4 suffer the most; each receives an increase of less than 1% in accepted sentences but must wait through 3.6 times the search. The trade-off seems even less favorable when we examine the difference in the number of roots produced by each system. The monolithic static grammar produces about five times as many roots on the average as a user's adapted grammar. Thus, in addition to slower response, the monolithic design requires more user interactions to arrive at a unique interpretation.

Turning to the results for users in the on-line experiments, we note that User 9 would notice little difference between interactions with a monolithic interface and those with CHAMP. User 10, on other hand, would notice

some difference, primarily in the number of interactions required to resolve the extra interpretations produced by the monolithic grammar. Most of the extra work required to accept User 10's sentences is caused by User 9's grammar. In other words, User 10 must pay for User 9's idiosyncracy in the monolithic design.

In essence, Figure 4-4 shows the price paid by each user for search over those portions of the monolithic system's language she will never use. We know that we cannot control for this kind of mismatch without sacrificing freedom of expression because of the real variability in preferred forms across users. Although it may be possible to achieve the same acceptance rate with somewhat less ambiguity in a much more carefully crafted monolithic grammar, the across-user variability is not going to disappear. Adaptation resolves this dilemma without requiring inordinate skill as a grammar designer. By learning the user's preferred forms of expression and engendering only those mismatches required by its small kernel, an adaptive interface keeps the cost of understanding each user in proportion to the inherent ambiguity in her idiosyncratic language.

Limited user adaptability. Figure 4-5 displays the results for the No-Adapt condition. Recall that the purpose of this variation is to study the claim that people adapt well enough to obviate the need for system adaptation. We measure user adaptation categorizing each utterance according to the minimum number of deviations required to interpret it. If the user adapted to the limitations of the system, we should see a general increase in *interpretable* sentences (those containing zero, one or two deviations).

The graph for User 6 does show an increase in interpretable sentences over her three sessions. Her behavior is interesting in two respects. First, it is reminiscent of User 1's behavior; User 6's grammar was fairly close to the kernel, especially with respect to vocabulary, and she tended to rely on terse forms. As a result, her utterances presented fewer loci for deviation than might occur in a more verbose style. Second, User 6, like the users in the Adapt condition, relied on those forms that had worked in the past. When an utterance met with failure, her next try was usually a minor variation that almost always succeeded. In short, User 6 performed successfully in the No-Adapt condition, but relatively little adaptation was required of her.

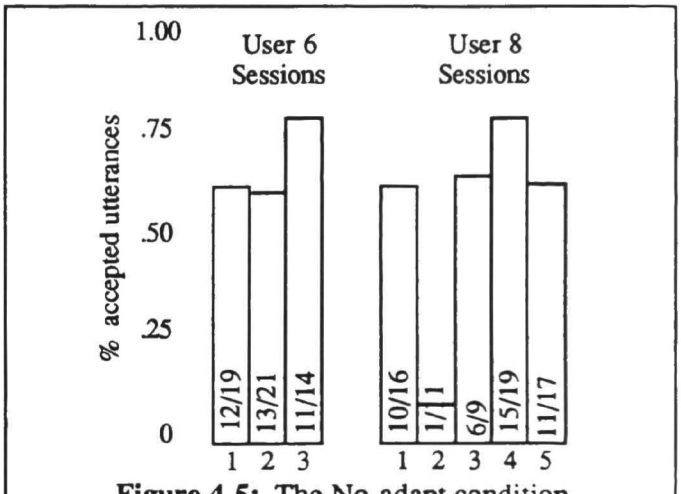


Figure 4-5: The No-adapt condition.

Figure 4-5 also displays User 8's performance in the No-Adapt condition. There is no pattern of improvement in User 8's data; her success at adapting to the interface's subset of English was extremely limited. Both her work experience and the content of her utterances lead to the conclusion that little of her behavior was attributable to task misunderstanding. Put simply, she was unable or refused to adapt. Her linguistic style, like that of User 3, is best characterized as verbose. In response to a typical utterance, the hidden operator told her which segments could be parsed and asked her to try again. She would then type exactly those segments just echoed, with no connecting text. When the terse form met with failure she would gravitate back to overly explanatory sentences.

User 8's performance in session two was so poor (in 31 minutes she produced one learnable sentence with two deviations) and her frustration so great, that she was given hints about how to use the system more effectively prior to beginning session three.[5] Although the help appeared to improve her performance in sessions three and four, in session five she still generated a high percentage of unparsable forms. This is in sharp contrast to the experiences of users in the Adapt condition, most of whom had no unparsable utterances by session five.

[5]Specifically, she was told to try to type simple but fully grammatical sentences and to think of the system as someone to whom she was writing instructions. She was the only user who was given this aid.

The contrast is sharpened by examining task completion. As mentioned above, all other users completed almost all tasks (98% completion on the average); User 8 completed only 44% of the tasks in her five sessions. This was due largely to the fact that she never managed to find an acceptable form of expression for an entire class of tasks (those involving the **change** action). Thus, although User 8 relied on the few forms that had worked in the past, her performance must be seen as a strong counterargument to the notion that everyone finds it natural or easy to adapt to a system's linguistic limitations.

5. Summary and Future Work

The purpose of these experiments was to establish certain behavioral characteristics of frequent users which, in turn, guarantee the conditions necessary for an adaptive interface to benefit user performance. The behavior of the users in the Adapt condition demonstrates the self-limiting, idiosyncratic language use predicted. As a result, we may assume that the conditions of within-user consistency and across-user variability will be met, making single-user, adaptive interfaces a practical and desirable alternative to a monolithic interface design.

The experimental results validate the idea of limited user adaptability as well. Although we cannot guarantee that a particular user will find adaptation to an interface's limitations difficult, User 8 serves as a dramatic example of the kind of limited user adaptability we must be prepared to encounter. Further, a comparison of Users 6 and 8 supports Watt's argument [11] that the ease with which a user adapts to a rigid interface depends significantly on a fortuitous correspondence between the user's natural language and the sublanguage provided by the interface designer. More importantly, the performance of the users in the adaptive condition demonstrates that an initial lack of correspondence *can* be overcome by system adaptation.

Despite these results, it remains to be seen whether the behavioral assumptions upon which adaptation relies hold over more realistic periods of time (nine months rather than nine sessions) and over very different tasks. Assuming such generality can be demonstrated, we believe adaptive parsing represents a significant step forward in the effort to provide users with full freedom of expression in computer interactions.

Acknowledgements: This research was partially supported by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 4976, and monitored by the Air Force Avionics Laboratory under Contract F33615-87-C-1499. The author was partially supported during this research by a Boeing Company fellowship. The content and clarity of this paper were improved by the comments of Angela Kennedy Hickman, Rick Lewis, and Thad Polk.

References

1. Good, M. D., Whiteside, J. A., Wixon, D.R., and Jones, J.J., "Building a User-Derived Interface," *Communications of the ACM*, Vol. 27, No. 10, 1984.
2. Harris, L. R., "Experience with ROBOT in 12 Commercial Natural Language Database Query Applications," *Proceedings of the Sixth International Joint Conference on Artificial Intelligence*, 1979.
3. Kelley, J. F., "An Iterative Design Methodology for User-Friendly Natural Language Office Information Applications," *ACM Transactions on Office Information Systems*, Vol. 2, No. 1, 1984.
4. Lehman, J. Fain, *Adaptive Parsing: Self-extending Natural Language Interfaces*, PhD dissertation, Carnegie Mellon University, 1989.
5. Lehman, J. Fain, "Adaptive Parsing: A General Method for Learning Idiosyncratic Grammars," *Proceedings of the Sixth International Conference on Machine Learning*, 1990.
6. Lehman, J. Fain, and Carbonell, J. G., "Learning the User's Language, A Step Towards Automated Creation of User Models," in *User Modelling in Dialog Systems*, Wahlster, W., and Kobsa, A., eds., Springer-Verlag, 1989.
7. Malhotra, A., "Knowledge-Based English Language Systems for Management Support: An Analysis of Requirements," *Proceedings of the Fourth International Joint Conference on Artificial Intelligence*, 1975.
8. Rich, E., "Natural Language Interfaces," *IEEE Computer*, Vol. 17, No. 9, 1984.
9. Slator, B. M., Anderson, M. P., and Conley, W., "Pygmalion at the Interface," *Communications of the ACM*, Vol. 29, No. 7, 1986.
10. Tennant, H., *Evaluation of Natural Language Processors*, PhD dissertation, University of Illinois, 1980.
11. Watt, W. C., "Habitability," *American Documentation*, July 1968.

THE COGNITIVE SPACE OF AIR TRAFFIC CONTROL: A PARAMETRIC DYNAMIC TOPOLOGICAL MODEL[1]

Steven Cushing

Boston University
Computer Science B-4
755 Commonwealth Avenue
Boston, MA 02215

Abstract

Recent observational work of controller behavior in simulations of air traffic control sessions suggests that controllers formulate and modify their plans in terms of *clusters* of aircraft, rather than *individual* aircraft, and that they cluster aircraft based on their *closeness in an abstract cognitive space*, rather than simple *separation in physical space*. A mathematical model of that space is presented as a background for further work to determine the *cognitive strategies* that controllers use to navigate that space. The model is *topological* in that *neighborhood constraints* play a central role; it is *dynamic* in that *more than one topology interact* to define its essential characteristics; and it is *parametric* in that *an entire class of spaces can be obtained* by varying the values of some parameters.

With the presented model as background, some hypotheses as to controller strategies are suggested and examples given to illustrate them. For example, controllers appear to segment their work into *episodes* defined in terms of the interactions of clusters and to *prioritize* the sub-tasks within these episodes, with different strategies for different sub-tasks. Some specific questions suggested by the hypotheses are raised, and some theoretical and practical implications are pointed out. For example, controllers appear to change their plans *consequent upon* changes in perceived clustering: *deliberate* cognitive acts are triggered by *presented* changes in conceptualization. This has implications for tool development, in that it underscores the need to limit the extent to which automated aids should be allowed to deviate from actual controller practice.

---

[1] Supported, in part, by Contract No. NGT 47-003-029, NASA-Langley Research Center, Hampton, VA. I would like to thank Herb Armstrong, Hugh Bergeron, Greg Bonadies, Randy Harris, Gary Lohr, Renate Rofske-Hofstrand, and George Steinmetz for invaluable assistance on the empirical work that underlies the theoretical model presented here. I also thank Eric Braude, Robert Kuhns, and especially Carol Munroe for very helpful comments on an earlier draft.

1. Background. Recent observational work on controller behavior in air traffic control simulation sessions suggests that controllers formulate and modify their plans in terms of *clusters* of aircraft, rather than *individual* aircraft, and that they cluster aircraft based on their *closeness in an abstract cognitive space*, rather than simple *separation in physical space* (Cushing, 1989; also see Bregman et al., 1988; Cushing, 1990; NASA/OAST, 1988; Roske-Hofstrand, 1988; Roske-Hofstrand et al., 1989; and Wesson, 1977; for further background and discussion). In this paper a mathematical model of that space is presented based on that work and as background for further work to determine the *cognitive strategies* that controllers use to navigate that space. The model is *topological* in that *neighborhood constraints* play a central role; it is *dynamic* in that *more than one topology interact* to define its essential characteristics; and it is *parametric* in that *an entire class of spaces can be obtained* by varying the values of some *external parameters*. The several topologies are also distinguished by the values of *internal parameters*. The parameters are presented in **Section 2** and the model itself in **Section 3**. With it as background, some hypotheses as to controller strategies are suggested in **Section 4**, along with some examples that illustrate them and some specific questions for further investigation.

2. The Parameters. The essence of the model is illustrated in Figures 1-9. Arriving aircraft are identified with (an initial segment of) the set $\mathbf{Z^+}$ of positive integers, and the controller's task is modelled as the construction of a permutation λ on $\mathbf{Z^+}$. In other words, aircraft that are presented in an order of arrival must be safely rearranged into an order for landing. The construction of an appropriate λ is subject to constraints that are central to the structure of the space. Aircraft that are within a neighborhood of each other in *arrival order* must be kept within a neighborhood of each other in *landing order*, with the respective neighborhoods defined by the parameters π_a and π_l. Similarly, aircraft that are within a neighborhood of each other in *arrival time* must be kept within a neighborhood of each other in *landing time*, with the respective neighborhoods defined by the parameters τ_a and τ_l. The topologies that are determined by these neighborhood constraints partially characterize the structure of the space.

Neighborhood constraints defined in terms of six other parameters determine further topologies that complete that characterization. Four of these, like π_a and π_l, are *numerical parameters*: the ***minimum separation*** (i.e., physical distance) σ that must be maintained at all times between any two aircraft, the ***maximum order difference*** ω and ***maximum cognitive distance*** κ for aircraft to be in a cluster, and the ***minimum duration interval*** ι that an aircraft can be in a cluster. In actual practice, σ has the value 3.0 (miles) for good weather and 5.0 for bad. A possible refinement would be to allow a further parameter σ_h for "heavy" aircraft, which require greater separation, but this can be avoided if the two separation values are taken to be systematically related, e.g., by a constant difference or ratio. The value of ι depends on short-term memory capacity and perception thresholds and can thus be expected, like ω and κ, to vary from controller to controller, in contrast to the values of σ, which are set by regulation.

Finally, we have two *functional parameters*: the ***physical distance*** ϕ and the ***cognitive distance*** χ . We can reasonably assume that ϕ is unique and fully understood, but determining the character of χ is intricately intertwined with the related problem of determining the cognitive strategies in which it is used and thus, other than being required to be a *distance metric*, is beyond the scope of the present paper.[2] Any of the parameters can be varied at will for purely investigative purposes, i.e., modeling, simulation, or the like.

3. The Model. Since we identify aircraft with their positions in arrival and landing order, i.e., before and after permutation by the controller, the parameters that bound their neighborhoods and clusters are naturally taken also to be integers. Physical or cognitive distances or times, however, are more reasonably taken to be real-valued. We thus begin as follows:

Stipulation: Numerical Parameters. Choose fixed values

$$\pi_a, \pi_l, \omega \in \mathbf{Z^+},$$
$$\tau_a, \tau_l, \sigma, \iota, \kappa \in \mathbf{R^+}.$$

Safety requires, first and foremost, that the *physical distance* between aircraft not be permitted to attain a value less than a specified minimum, identified above as the ***minimum separation*** σ . It is apparent from controller behavior, however, that a very different measure of *cognitive distance* also plays a role in determining which aircraft are taken as being "close enough" to be considered together as a ***cluster***, as illustrated in Figures 4 and 5. Even aircraft whose arrival occurs at opposite ends of the controller's screen and are thus widely separated physically can be grouped together in planning, as the required permutation is constructed. We thus continue as follows:

Stipulation: Functional Parameters. Choose distinct fixed values

$$\phi, \chi : \mathbf{Z^+ \times Z^+ \times R^+ \to R},$$

subject to the following constraint:

For $\delta \in \{\phi, \chi\}$, $\forall\ p_1, p_2, p_3 \in \mathbf{Z^+}$, $t \in \mathbf{R^+}$,

$$0 \le \delta(p_1, p_2, t),$$
$$\delta(p_1, p_2, t) = \delta(p_2, p_1, t),$$
$$\delta(p_1, p_2, t) \le \delta(p_1, p_3, t) + \delta(p_3, p_2, t).$$

---

[2] Compare the analogous difference, in physics, between claiming that the universe is a four–dimensional differentiable manifold and claiming that it is a *particular* such manifold or, in linguistics, between claiming that language is characterized by an underlying autonomous universal syntax and claiming that that syntax is embodied in grammatical rules or principles *of a particular form*. Though equally non-trivial, the two sorts of claims differ substantially in strength, the first providing the background for investigating the second, further refinements of which can eventually lead to the confirmation of both (or not). The space proposed here embodies a claim of the former sort; further characterizing χ would involve successive refinements of claims of the latter.

This is the standard definition of ***distance metric*** in a metric space, but *relativized* to times and with an *internal parameter* δ that takes the *two* functions ϕ and χ as instances, thereby defining two different topologies on the space. Note that the stipulated relativization allows δ itself to vary with time, with different results possible at different times for the same ordering or spatial arrangement of aircraft. Though not sufficiently constraining for *physical* distance, this flexibility is desirable for *cognitive* distance, which can be expected to vary with time of day, work load, stress, and the like, as these affect the controller's mental state.

The *difference in arrival order*, $p_2 - p_1$, is a further distance metric, *not* dependent on time, that, along with χ, constrains the formation of clusters. *It is the occurrence of clustering that makes controller cognition a non-trivial problem.* If aircraft were treated entirely as individuals, it would be a simple matter, at least in principle, to automate the process, by assigning each aircraft a dedicated processor to maintain separation σ from every other, as is apparent from the Figures. With each aircraft capable of fending for itself, air traffic control would be rendered superfluous. Clustering reflects the need, however, *to plan ahead* in maintaining σ, not just *now*, but *throughout* the permutation process. It is thus in the internal logic of clusters that we can most reasonably expect to find indications of how this planning proceeds.

Definition: Neighborhood. Let $\mathbf{D} \in \{\mathbf{Z^+}, \mathbf{R^+}\}$. $\forall\ d_1, d_2, \beta \in \mathbf{D}$,
d_2 **is in a** β**–neighborhood of** d_1, $\mathbf{N}_\beta (d_1, d_2)$, if $|d_2 - d_1| < \beta$.

Corollary: **(1)** **Reflexive.** $\forall\ d, \beta \in \mathbf{D}, \mathbf{N}_\beta (d, d)$.

(2) **Symmetric.** $\forall\ d_1, d_2, \beta \in \mathbf{D}, \mathbf{N}_\beta (d_1, d_2) \Rightarrow \mathbf{N}_\beta (d_2, d_1)$.

(3) **Intransitive.** $\exists\ d_1, d_2, d_3, \beta \in \mathbf{D}, \mathbf{N}_\beta (d_1, d_2) \wedge \mathbf{N}_\beta (d_2, d_3) \wedge \neg \mathbf{N}_\beta (d_1, d_3)$.

Again, this is a standard definition of *neighborhood* in a metric space, but with the ***domain*** **D** and the ***neighborhood bound*** β as parameters *internal to the model*, taking $\mathbf{Z^+}$ or $\mathbf{R^+}$ and the external parameters π_a, π_l, τ_a, τ_l, or ω as values, respectively, in various places in the model. The corollary is also standard and serves mainly to disqualify $\mathbf{N}_\beta$ from being an *equivalence relation*.

Definition: Clustering. A function $\mathbf{c}: \mathbf{Z^+ \times R^+} \rightarrow \mathbf{2}^{\mathbf{Z^+}}$ is a $(\omega, \iota, \kappa, \chi)$**–clustering** if it satisfies the following constraints:

(1) (a) $\forall\ p \in \mathbf{Z^+}, \exists\ t_{p,a}, t_{p,l} \in \mathbf{R^+}, \forall\ t \in \mathbf{R^+}$,
$[t \in [t_{p,a}, t_{p,l}] \Rightarrow p \in \mathbf{c}(p, t)] \wedge [t \notin [t_{p,a}, t_{p,l}] \Rightarrow \mathbf{c}(p, t) = \varnothing]$,

(b) $\forall\ p_1, p_2 \in \mathbf{Z^+}, \forall\ t \in \mathbf{R^+}, [p_2 \in \mathbf{c}(p_1, t) \Rightarrow p_1 \in \mathbf{c}(p_2, t)]$,

(c) $\forall\ p_1, p_2, p_3 \in \mathbf{Z^+}, \forall\ t \in \mathbf{R^+}$,
$[p_2 \in \mathbf{c}(p_1, t) \wedge p_3 \in \mathbf{c}(p_2, t)] \Rightarrow p_3 \in \mathbf{c}(p_1, t)$,

(2) (a) $\forall\ p_1, p_2 \in \mathbf{Z^+}, t \in \mathbf{R^+}, [p_2 \in \mathbf{c}(p_1, t) \Rightarrow \mathbf{N}_\omega(p_1, p_2)]$,

(b) $\forall\ p_1, p_2 \in \mathbf{Z^+}, t \in \mathbf{R^+}, [p_2 \in \mathbf{c}(p_1, t) \Rightarrow \chi(p_1, p_2, t) < \kappa]$,

(3) $\forall\ p \in \mathbf{Z^+}, t \in \mathbf{R^+}$,

$[\mathbf{c}(p, t) \neq \varnothing \Rightarrow [\exists\ t_1, t_2 \in \mathbf{R^+}, t_2 - t_1 \geq \iota, t \in [t_1, t_2],$

$\forall\ t' \in (t_1, t_2), [\mathbf{c}(p, t') \neq \varnothing \Rightarrow \mathbf{c}(p, t') = \mathbf{c}(p, t)]\ ]\ ]$.

A *clustering* assigns to *each* aircraft (in $\mathbf{Z^+}$) a *set* of aircraft, its *cluster*, (in $\mathbf{2^{Z^+}}$) that can *change* as time (in $\mathbf{R^+}$) passes, subject to three constraints. Constraint **(1)** says that ***being-clustered-with***, unlike ***being-in-a-neighborhood-of***, is an *equivalence relation* at a time t on sets of aircraft with overlapping ***salience intervals*** $[t_{p,a}, t_{p,l}]$: an aircraft is always in its own cluster between its arrival time and its landing time, but has *no* genuine cluster (i.e., its cluster = $\varnothing$) otherwise. Outside that interval, an aircraft is of no cognitive interest, in the present context; it need not be worried about before it arrives and after it lands. Assigning it a cluster of $\varnothing$ simplifies the formulations by avoiding the need for partial functions. An aircraft can be viewed as *comprising* its own cluster whenever it is treated individually. Constraint **(2)** relates clustering to the two *non-physical distance metrics* of the model by imposing *proximity constraints*: one aircraft can be in the cluster of another aircraft *only* if they are within ω of each other in ***arrival order*** and within κ of each other in ***cognitive distance***. The actual values of ω and κ will depend on the individual controller, but they can be expected to be intimately connected to limits on short-term memory. *Characterizing* cognitive distance amounts to *solving simultaneously the conditions of the definition* for suitable functions χ .[3] Constraint **(3)** says that a clustering can assign a particular set of aircraft as the cluster of a particular aircraft at a salient time *only* if it assigns that set as the cluster of that aircraft *within an interval that contains that time*, of width no less than ι. This guarantees *a degree of* (i.e., step-wise) *continuity* so that *changes* in clustering remain realistic, without diminishing the flexibility that makes those changes useful.

The parameters ω and ι serve to define upper bounds on the size and the number of clusters and on the number of times they can change, as follows:

Theorem: Let $\boldsymbol{C}$ be the set of all $(\omega, \iota, \kappa, \chi)$-clusterings.

(1) $\forall\ p \in \mathbf{Z^+}, t \in \mathbf{R^+}, \mathbf{c} \in \boldsymbol{C}$,

$|\mathbf{c}(p, t)| \leq 2\omega - 1$.

(2) $\forall\ p \in \mathbf{Z^+}, t \in [t_{p,a}, t_{p,l}]$,

$|\{\mathbf{c}(p, t) \mid \mathbf{c} \in \boldsymbol{C}\}| \leq 2^{2(\omega - 1)}$.

(3) $\forall\ p \in \mathbf{Z^+}, \mathbf{c} \in \boldsymbol{C}$,

$|\{t \mid t \in [t_{p,a}, t_{p,l}), \exists\ \varepsilon \in \mathbf{R^+}, \mathbf{c}(p, t - \varepsilon) \neq \mathbf{c}(p, t + \varepsilon)\}|$

$< ((t_{p,l} - t_{p,a}) / \iota) + 2$.

[3] See note 2.

Proof: (1) Planes available for p's cluster range from $p - (\omega - 1)$ through $p + (\omega - 1)$, including p itself:
$\therefore\ (\omega - 1) + (\omega - 1) + 1 = 2\omega - 1.$

(2) Clusters generally available for p include *all subsets* of the set of planes in (1) *that contain p*:
$\therefore\ (1/2)(2^{2\omega - 1}) = 2^{2\omega - 2} = 2^{2(\omega - 1)}.$
The $\leq$ is required, rather than $=$, because of the current indeterminacy of χ, which can be expected, when determined, to rule out possible clusters that are permitted by the other constraints, and also, independently of χ, because of the fact that two different planes will not have *identical* salience intervals (see ***landing sequence*** below).

(3) A plane's cluster becomes non-null upon arrival and can then change at any time thereafter. After the first change, each cluster must then persist for an interval of at least ι in duration. Let $L = (t_{p,l} - t_{p,a})$ and assume each interval has duration exactly ι. If ι exactly divides L, there are L / ι change points if $t_{p,a}$ begins an interval of length ι, and $(L / \iota) + 1$ otherwise. If ι does not exactly divide L, there are $[(L / \iota) + 1]$ change points if $t_{p,a}$ begins an interval of length ι, and $[(L / \iota) + 2]$ otherwise, where $[x]$ is the greatest integer in x. If one or more intervals has duration greater than ι, the number of change points does not increase.

In other words, the number *of distinct aircraft* that are *in the cluster of a particular aircraft at a particular time* is bounded above by $2\omega - 1$; the number *of distinct potential clusters* that are available to be associated *with a particular aircraft at a particular time, between the time it arrives and the time it lands,* is bounded above by $2^{2(\omega - 1)}$; and the number *of distinct times* that a controller *assigns a new* ***non-null*** (hence the "**)**" in $[t_{p,a}, t_{p,l})$ here) *cluster to a particular aircraft* is bounded above by $((t_{p,l} - t_{p,a}) / \iota) + 2$.

Finally, we can characterize ***the controller's task*** as follows:

Definition: Landing Sequence. A permutation λ on $\mathbf{Z^+}$, i.e., a bijection $\lambda : \mathbf{Z^+} \rightarrow \mathbf{Z^+}$, is a $(\pi_a, \pi_l, \tau_a, \tau_l)$**–landing sequence** if it satisfies the following constraint:

$$\forall\ p_1, p_2 \in \mathbf{Z^+},$$
$$[\,[\mathbf{N}_{\pi_a}(p_1, p_2) \Rightarrow \mathbf{N}_{\pi_l}(\lambda\ (p_1), \lambda\ (p_2))]$$
$$\wedge\ [\mathbf{N}_{\tau_a}(t_{p_1,a}, t_{p_2,a}) \Rightarrow \mathbf{N}_{\tau_l}(t_{p_1,l}, t_{p_2,l})]$$
$$\wedge\ [p_1 \neq p_2 \Rightarrow t_{p_1,l} \neq t_{p_2,l}]\,].$$

Controller's Task: Construct a clustering and a landing sequence subject to the following constraint:

$$\forall\ p_1, p_2 \in \mathbf{Z^+},\ t \in [t_{p_1,a}, t_{p_1,l}] \cap [t_{p_2,a}, t_{p_2,l}],\quad \phi(p_1, p_2, t) \geq \sigma\ .$$

In other words, aircraft that are "near" each other in *arrival* order or time must be "near" each other in *landing* order or time, respectively, though the "nearness" *criteria* for arrival and landing need not be the same and the landing *times* must be different. Furthermore, no matter how they get permuted or what clustering is used, two aircraft must never be permitted to approach each other to within a physical distance of σ. Constructing a landing sequence is the *explicit* requirement of the controller's task; the need to construct a clustering arises from the inherent complexity of that task in the context of human cognition.

4. Further Work in Progress. The above model provides the *general cognitive framework* within which controllers appear to do their work. The next task is to determine the cognitive *strategies* that controllers use to navigate the space the model characterizes. Work thus far has suggested the following hypotheses (Cushing, 1989):

(1) Controllers segment their work temporally and dynamically into (*sometimes overlapping*) *episodes* and *sub-episodes* defined in terms of the interactions of aircraft clusters;
(2) Controllers *prioritize* the sub-tasks within their episodes, with different strategies for different sub-tasks; and
(3) Controllers change plans *consequent upon* changes in perceived clustering: *deliberate* cognitive acts are triggered by *presented* changes in conceptualization.

Hypothesis (1) is illustrated throughout the Figures. As an example of hypothesis (2), a controller in a simulation session checks that separation of aircraft is adequate both before and after doing an artificial "side–task" consisting of reading extraneous information about the weather, but is less thorough in checking before scanning to accept responsibility for an aircraft that is being handed off to him by another controller. This suggests that he considers the latter task to be more important and in need of more immediate attention when it arises. Hypothesis (3) has particular implications for tool development, in that it underscores the need to limit the extent to which *automated aids*, such as expert system or decision support tools, should be allowed to deviate from actual controller practice. Further work is needed to quantify and test these hypotheses.

In particular, the following questions need to be answered: (1) What factors *other than arrival order* play a role in clustering?[4] (2) Does the controller check aircraft separation *in preparation* for doing the "side-task," or does he do the "side-task" *after having checked* separation? (3) To what extent does the controller maintain separation *of clusters*, and to what extent is he willing to *shuffle* (i.e., modify and mix) them? (4) How often and why does the controller scan *back* to aircraft that are *already* lined up for landing, while focusing primarily on a *later* cluster, and how often and why does he scan to *outliers* beginning a *new* cluster, while focusing primarily on an earlier one?

It is anticipated that the pursuit of answers to these and related questions will lead to refinements in the above model, as the strategies it supports are unraveled. It will be necessary to investigate the extent to which controllers *differ* in their choice of strategies, both from each other and from non–controller control subjects, in order to determine the extent to which the strategies used are *learned*, rather than corollaries of *inherent* properties of human perceptual and cognitive mechanisms. How this question gets answered has implications for *training methods*, even aside from the development of support tools for assisting controllers on the job.

[4] I.e., just what is it that distinguishes **(b)** from **(a)** in condition **(2)** of the clustering definition?

References

Bregman, Howard L.; McCabe, Warren L.; and Sutcliffe, William G. 1988. "Capturing Air Traffic Controller Expertise for Incorporation in Automated Air Traffic Control Systems." ***Proceedings of the Human Factors Society, 32nd Annual Meeting.*** pp. 1031-1035.

Cushing, Steven. 1989. "From Where They Look to What They Think: Determining Controllers' Cognitive Strategies from Oculometer Scanning Data." In Tiwari, Surendra N. (Comp.). NASA Contractor Report 181894. Contract No. NGT 47-003-029. NASA-Langley Research Center, Hampton, VA. September 1989.

Cushing, Steven. 1990. Letter. ***The Sciences.*** **30(1)**:14. January/February 1990.

NASA/OAST. 1988. "Aviation Safety/Automation: A Plan for the Research Initiative." National Aeronautics and Space Administration, Office of Aeronautics and Space Technology. November, 1988.

Roske-Hofstrand, Renate. 1988. "Discourse Understanding of Displayed Actions: An Investigation of the Temporal Relations of Event Schemas and their Effect on Reconstruction." Doctoral Dissertation. New Mexico State University, Las Cruces.

Roske-Hofstrand, Renate; Armstrong, Herbert; and Bergeron, Hugh. 1989. "Applied Cognitive Research in Air Traffic Control: A Concept Paper and Research Plan." Draft Report. NASA-Langley Research Center, Hampton, VA.

Wesson, Robert Bell. 1977. "Problem-solving with Simulation in the World of an Air Traffic Controller." Doctoral Dissertation. University of Texas, Austin. N78–27098. University Microfilms Dissertation Information Service.

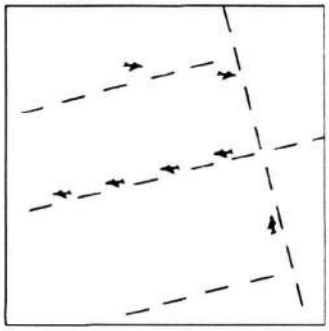


Figure 1: A typical ATC *arrival* situation: Aircraft arriving from *several directions* must be *lined up* for landing.

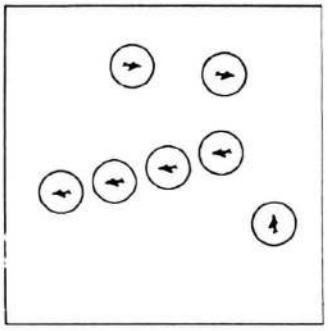


Figure 2: *Minimum separation* must be maintained between *any two aircraft* at *any time*.

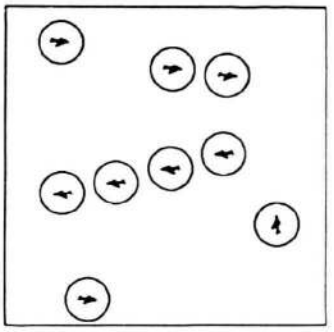


Figure 3: *Distant, but contemporaneous*, arrivals are treated together in *planning*.

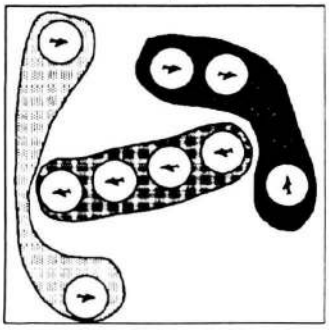


Figure 4: Aircraft are *clustered* by "closeness" in an abstract *cognitive* space, rather than in actual *physical* space.

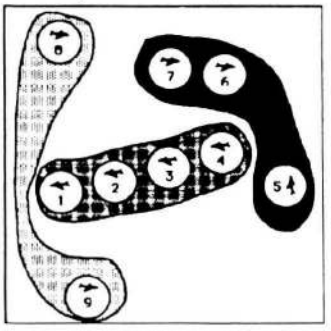


Figure 5: *Cognitive "closeness"* is measured by a *spatio-temporal metric* distinct from *arrival order* and *physical separation*.

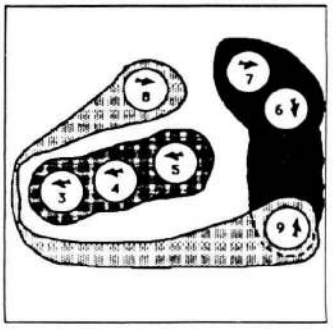


Figure 6: Clusterings can *change*, due to *unforeseen circumstances*: Aircraft can *leave* one cluster and *enter* another.

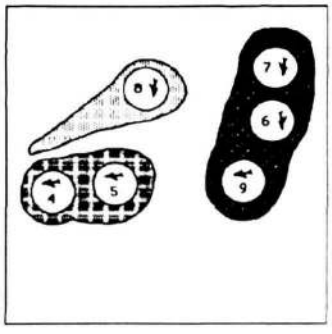


Figure 7: ATC space has a *dynamic topology* that reflects the *interaction* of *multiple factors*.

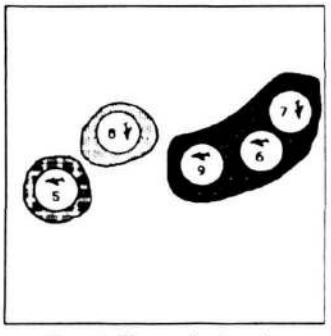


Figure 8: Changes in *clustering* correlate with changes in *plan*, in *response* to new circumstances.

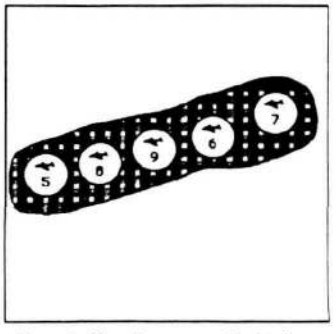


Figure 9: Aircraft *merge* on final to form a single cluster: The *landing* order is a *permutation* of the *arrival* order.

MODELS OF NEUROMODULATION AND INFORMATION PROCESSING DEFICITS IN SCHIZOPHRENIA

David Servan-Schreiber and Jonathan D. Cohen

School of Computer Science and Department of Psychology
Carnegie Mellon University
and
Western Psychiatric Institute and Clinic
University of Pittsburgh

Abstract

This paper illustrates the use of connectionist models to explore the relationship between biological variables and cognitive deficits. The models show how empirical observations about biological and psychological deficits can be catpured within the same framework to account for specific aspects of behavior. We present simulation models of three attentional and linguistic tasks in which schizophrenics show performance deficits. At the cognitive level, the models suggest that a disturbance in the processing of context can account for schizophrenic patterns of performance in both attention and language-related tasks. At the same time, the models incorporate a mechanism for processing context that can be identified with the function of the prefrontal cortex, and a parameter that corresponds to the effects of dopamine in the prefrontal cortex. A disturbance in this parameter is sufficient to account for schizophrenic patterns of performance in the three cognitive tasks simulated. Thus, the models offer an explanatory mechanism linking performance deficits to a disturbance in the processing of context which, in turn, is attributed to a reduction of dopaminergic activity in prefrontal cortex.

Schizophrenia is marked by a wide variety of behavioral deficits, including disturbances of attention, language processing and problem solving. Schizophrenia is also characterized by biological abnormalities, including disturbances in specific neurotransmitter systems (e.g., dopamine and norepinephrine) and anatomic structures (e.g., prefrontal cortex and hippocampus). For the most part, however, the behavior and biology of schizophrenia have remained separate fields of inquiry. Despite a modern consensus that information processing deficits in schizophrenia are the result of underlying biological abnormalities, few efforts have been made to specify exactly how these phenomena relate to one another.

In this paper we address this issue by drawing upon the framework of connectionist models. This framework provides theoretical concepts that are intermediate between the details of neuroscientific observations and the box-and-arrow diagrams of traditional information processing or neuropsychological theories. We explore the ability of connectionist models to explain aspects of schizophrenic behavior in terms of specific underlying biological disturbances. At the behavioral level, the models address both normal and schizophrenic performance in three experimental tasks: two that tap attentional performance (the Stroop task and the continuous performance test), and one that measures language processing abilities (a lexical disambiguation task). The models use a common set of information processing mechanisms, and show how a number of seemingly disparate observations about schizophrenic behavior can all be related to a single functional deficit: a disturbance in the processing of context.

The models also suggest that this functional deficit may be explained by a specific biological disturbance: a reduction in the effects of the neurotransmitter dopamine in prefrontal cortex (PFC). First, we show how a particular parameter of the models can be used to simulate the neuromodulatory effects of dopamine at the neuronal level. We then present the results of simulations in which this parameter is disturbed within a module corresponding to the function of PFC. In each of the three behavioral simulations, this disturbance leads to changes in performance that quantitatively match those observed for schizophrenics in the corresponding tasks. These findings suggest that a number of the disturbances of attention and language found in schizophrenia may all result from a disturbance in the processing of context which, in turn, may be explained by a single biological abnormality, a reduction of dopaminergic activity in PFC.

The background of the models presented in this paper spans a large and diverse literature concerning cognitive deficits in schizophrenia, the anatomy and physiology of dopamine systems, the neurophysiology and

---

The order of authorship was determined by the toss of a coin. This work was supported by a NIMH Physician Scientist Award (MH00673) to JDC and a NIMH Individual Fellow Award (MH09696) to DSS. Correspondence concerning this paper should be addressed to either author at Carnegie Mellon University, Pittsburgh, PA 15213.

neuropsychology of frontal cortex, and the role of these biological systems in schizophrenia. A full review of these data is beyond the scope of this communication (see Cohen and Servan-Schreiber, 1989, for a more comprehensive review). Here, we highlight five empirical observations concerning information processing and biological deficits in schizophrenia. The purpose of our simulations is to show how these observations — which range from the biological to the cognitive level — can be articulated within the same model and account for specific aspects of behavior. These observations are: 1) Schizophrenics' performance in a variety of cognitive tasks indicates a decreased ability to use context information for selecting appropriate behavior. By *context*, we mean information that is relevant to, but is not part of the content of a behavioral response. This can be task instructions or specific previous stimuli that determine correct behavior. 2) Prefrontal cortex (PFC) is directly involved in, and necessary for the representation and maintenance of context information. 3) The normal function of PFC relies on the activity of the mesocortical dopamine system. 4) Dopamine has a modulatory effect on the activity of PFC, by influencing the responsivity, or *gain*, of cells in this brain region; and 5) Schizophrenia is associated with abnormalities of both frontal cortex and dopamine activity.

Disturbances in the Processing of Context in Schizophrenia

The Stroop task. This task taps a fundamental attentional phenomenon: the ability to respond to one set of stimuli, even when other, more compelling stimuli are available. The paradigm consists of two subtasks. In one, subjects name the color of the ink in which a word is printed. In the other, subjects read the word aloud while ignoring ink color. Normal subjects are less able to attend selectively to colors (i.e., ignore words) than the reverse. If schizophrenics suffer from an attentional deficit then this effect should be exacerbated; that is, they should be less able to ignore word information, and should show a greater interference effect. This prediction is supported by studies of schizophrenic performance in the Stroop task (Abramczyk et al., 1983; Wapner & Krus, 1960). However, because an overall slowing of reaction time is also observed, the significance of an increase in interference has been called into question: This may simply reflect an unanticipated effect of general slowing of performance, rather than of a specific attentional deficit. This issue has not been resolved in the literature. Below, we will show how a simulation model of this task can help distinguish the effects of a general slowing from those of a specific attentional deficit.

Considerations of the Stroop effect typically focus on the role of selective attention. However, the processing of context is also central to this task. In order to respond to the appropriate dimension of the stimulus, the subject must hold in mind the task instructions for that trial. These provide the necessary context for interpreting the stimulus and generating the correct response. In Stroop experiments, trials are typically blocked by task (e.g., all color naming, or all word reading), so that the proper context is consistent, and regularly reinforced. However, in other attentional tasks — such as the continuous performance test — this is not the case.

The Continuous Performance Test. In this task (CPT), subjects are asked to detect a target event among a sequence of briefly presented stimuli, and to avoid responding to distractor stimuli. The target event may be the appearance of a single stimulus (e.g, detect the letter 'X' appearing in a stream of other letters), or a stimulus appearing in a particular context (e.g, respond to 'X' preceded by 'A'). Schizophrenics (and often their biological relatives) are typically impaired in their ability to discriminate between target and distractors on this task, compared to normal and patient controls (e.g., Kornetsky, 1972; Nuechterlein, 1984). This deficit is most apparent in versions of the task that make high processing demands. For example, in the 'CPT-double' a target event consists of two consecutive identical letters. Memory for the previous letter provides the context necessary to evaluate the significance of the current letter; inability to use this context would impair performance. Schizophrenics perform especially poorly in this and similar versions of the task.

Schizophrenic Language Deficits. Schizophrenics also show poor use of context in language processing. Chapman et al. (1964) first demonstrated this in a study of schizophrenics' interpretation of lexical ambiguities. They found that schizophrenics tended to interpret the strong (dominant) meaning of a homonym used in a sentence, even when context suggested the weaker (subordinate) meaning. For example, given the sentence "The farmer needed a new *pen* for his cattle," schizophrenics interpreted the word "pen" to mean writing implement more frequently than control subjects. They did not differ from control subjects in the number of unrelated meaning responses that were made (e.g., interpreting "pen" to mean "fire truck"), nor did they differ in the number of errors made when the strong meaning was correct. Recently, we tested the idea that schizophrenics are restricted in the *temporal range* over which they can process linguistic context (Cohen et al., 1989). We designed a lexical ambiguity task, similar to the one used by Chapman and his colleagues, in which we could manipulate the temporal parameters involved.

Subjects were presented with sentences made up of two clauses; each clause appeared one at a time on a computer screen. One clause contained an ambiguous word in neutral context (e.g., "you need a PEN"), while the other clause provided disambiguating context (e.g., "in order to keep chickens" or "in order to sign a check"). Clauses were designed so that they could be presented in either order: context first or context last. The ambiguity in each sentence always appeared in capital letters, so that it could be identified by the subject. Subjects were presented with

the sentences and, following each, were asked to interpret the meaning of the ambiguity as it was used in the sentence. Sentences were distributed across three conditions: a) *weak* meaning correct, context *last*; b) weak meaning correct, context *first*; c) *strong* meaning correct, context first. For example, a given subject would have seen the ambiguity "pen" in one of the three following conditions, and then chosen their response from the list of possible meanings:

(a) you can't keep chickens [*clear screen / pause*] without a PEN (*weak meaning, context first*)

— *or* —

(b) without a PEN [*clear screen / pause*] you can't keep chickens (*weak meaning, context last*)

— *or* —

(c) you can't sign a check [*clear screen / pause*] without a PEN (*strong meaning, context first*)

[*clear screen / pause*]

The meaning of the word in capital letters is:

a writing implement	a fenced enclosure	a kind of truck
(*strong meaning*)	(*weak meaning*)	(*unrelated meaning*]

The results of this study (shown in Figure 7) corroborated both the Chapmans' original findings, and the explanation of their findings in terms of an inability to maintain context. Thus, as the Chapmans found, schizophrenics made significantly more dominant meaning errors than did controls when the weak meaning was correct. However, this only occurred when the context came *first* (i.e., less recently — condition *B* above). When context came last (i.e., more recently), schizophrenics correctly chose the weak meaning. This was the only type of error that reliably distinguished schizophrenics from controls. These findings suggest that the impairment observed in language tasks may be similar in nature to the impairments observed in attentional tasks: a difficulty in representing and maintaining context.

PFC, Context, and Dopamine

Several studies suggest that frontal areas are specifically involved in maintaining context information for the control of action. For example, at the neurophysiological level, Fuster (1980) and Goldman-Rakic (1985) have observed cells in PFC that are specific to a particular stimulus and response, and that remain active during a delay between these. They have argued that neural patterns of activity are maintained in PFC which encode the temporary information needed to guide a response. At the behavioral level, these authors and Diamond (e.g., 1989) have also reported data showing that PFC is needed to perform tasks involving delayed responses to ambiguous stimuli. Diamond has emphasized that prefrontal memory is required, in particular, to overcome reflexive or previously reinforced response tendencies in order to mediate a contextually relevant — but otherwise weaker — response.

Furthermore, it has been shown that dopaminergic innervation of PFC is necessary for this brain region to maintain contextual information. Experimental lesions in animals, or clinical lesions in humans, to this dopaminergic supply can mimic the effect of lesions to the PFC itself on behavioral tasks requiring memory for context (e.g., Brozoski et al., 1979).

Neuromodulatory Effects of Dopamine

Several anatomical and physiological observations support the idea that catecholamines such as dopamine and norepinephrine modulate information processing in the brain. Dopamine and norepinephrine neurons originate in discrete nuclei localized in the brain stem and their fibers project radially to several functionally different areas of the CNS. The baseline firing rate of these neurons is low and stable, and the conduction velocity along their fibers is slow. These characteristics result in a steady state of transmitter release and relatively long-lasting post-synaptic effects that are consistent with a modulatory role. Most importantly, recent evidence suggests that the effect of dopamine release is not to directly increase or reduce the firing frequency of target cells (e.g., Chiodo and Berger, 1986). Rather, like norepinephrine, dopamine seems to modulate the response properties of post-synaptic cells such that both inhibitory and excitatory responses to other afferent inputs are *potentiated*. This effect has been described as an increase in the 'signal-to-noise ratio' of the cells' behavior or an 'enabling' of its response (e.g., Foote et al., 1975) .

PFC and Dopamine in Schizophrenia

The behavioral data reviewed earlier concerning schizophrenic performance deficits indicates an insensitivity to context, and a dominant response tendency. This is consistent with evidence that schizophrenia is associated with frontal lobe impairment. Schizophrenics show typical frontal lobe deficits on standard neuropsychological tests, including the Wisconsin Card Sort Test (WCST) (e.g., Malmo, 1974) and the Stroop task (as described above). In addition, imaging and electrophysiological studies suggest an atrophy and abnormal metabolism in the frontal lobes

of schizophrenics (e.g., Ingvar and Franzen, 1974). Recent studies have even demonstrated abnormal metabolism in the PFC of schizophrenics specifically during performance on tasks requiring memory for context such as the Wisconsin Card Sort Task and a variant of the CPT (Weinberger et al., 1986; R.M. Cohen et al., 1987). This work confirms that anatomic and physiological deficits of frontal cortex may indeed be associated with some behavioral deficits observed in schizophrenics.

Frontal lobe dysfunction in schizophrenia fits well with the prevailing neurochemical and psychopharmacological data concerning this illness. The PFC is a primary projection area for the mesocortical dopamine system, a disturbance of which has consistently been implicated in schizophrenia (e.g., Meltzer & Stahl, 1976). In view of these findings, several authors have proposed that reduced dopaminergic tone in PFC may be associated with hypofrontality in schizophrenia, and may be responsible for several of the cognitive deficits that have been observed (e.g., Weinberger & Berman, 1988).

Summary

We referred to evidence that schizophrenics inadequately maintain context for the control of action; that the PFC plays a role in maintaining context; that an intact mesocortical dopamine system is necessary for normal PFC function; and finally, that the mesocortical dopamine system is affected in schizophrenia. Despite a growing recognition that these observations are related, no theory has yet been proposed which explains — in terms of causal mechanisms — the relationship between disturbances in PFC and dopamine on the one hand, and behavioral deficits on the other. In the remainder of this paper, we present a set of connectionist models that simulate schizophrenic performance in the tasks described above.

Simulation of the Physiological Effects of Dopamine

In the models, the action of dopamine is simulated as a change in a parameter of the function relating a unit's input to it activation value. To do so, we first assume that the relationship between the input to a neuron and the neuron's frequency of firing can be simulated as a non-linear function relating the net input of a model unit to its activation value. Physiological experiments suggest that in biological systems the shape of this function is sigmoid, with its steepest slope around the baseline firing rate (e.g., Freeman, 1979). The same experiments also indicate that small increments in excitatory drive result in greater changes in firing frequency than equivalent increments in inhibitory input. These properties can be captured by the logistic function with a constant negative bias:

$$\text{activation} = \frac{1}{1+e^{-(\text{gain}*\text{net})+\text{bias}}} \quad \text{(see Figure 1, Gain = 1.0)}$$

The potentiating effects of dopamine can be simulated by increasing the gain parameter of the logistic function. As Figure 1 (Gain = 2.0) illustrates, with a higher gain the unit is more sensitive to afferent signals, while its baseline firing rate (net input = 0) remains the same. Elsewhere, we have shown that such a change in gain can simulate a number of different catecholaminergic effects at both the biological and behavioral levels (e.g., Servan-Schreiber, Printz & Cohen, 1990).

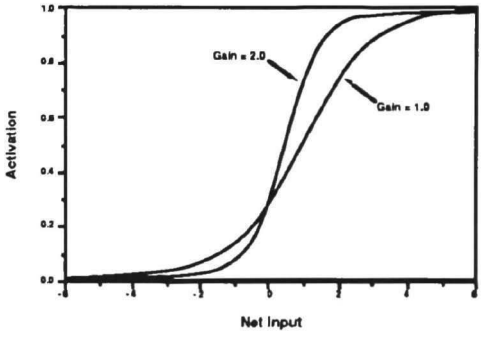


Figure 1

In order to simulate the effect of a neuromodulator, we change gain equally for all units in the model that are assumed to be influenced by that neuromodulator. For example, the mesocortical dopamine system has extensive projections to prefrontal cortex. To model the action of dopamine in this brain area, we change the gain of all units in the module corresponding to this area. In the models described below, decreased dopamine activity in prefrontal cortex was simulated by reducing the gain of units in the module used to represent and maintain context. In all three models, simulation of schizophrenic performance was conducted by reducing gain from a normal value of 1.0 to a lower value in the range 0.6-0.7.

Simulation of the Stroop effect

Elsewhere, we have described a connectionist model of selective attention that simulates human performance in the Stroop task (Cohen, Dunbar & McClelland, in press). In brief, this model consists of two processing pathways, one for color naming and one for word reading and a task demand module that can selectively facilitate processing in either pathway (see Figure 2) Simulations are conducted by activating input units corresponding to stimuli used in an actual experiment (e.g., the input unit in the color naming pathway representing the color red) and the appropriate task demand unit. Activation is then allowed to spread through the network. This leads to activation of the output unit corresponding to the appropriate response (e.g., "red").

This simple model is able to simulate an impressive number of empirical phenomena associated with the Stroop task. It captures the four basic effects (asymmetry in speed of processing between word reading and color naming, the immunity of word reading to the effects of color, the susceptibility of color naming to interference and facilitation from words (and greater interference than facilitation), as well as the influence of practice on interference and facilitation effects, the relative nature of these effects, response set effects and stimulus onset asynchrony effects (see Cohen et al., in press).

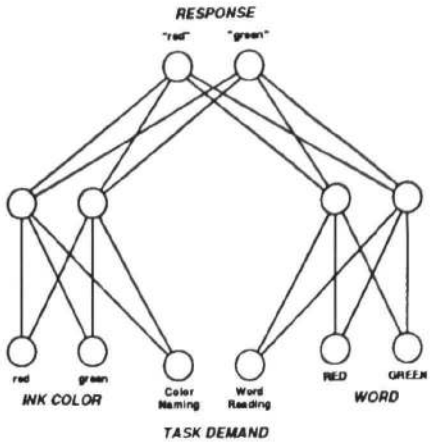


Figure 2

This model also exhibits behaviors that make it relevant to understanding schizophrenic disturbances of attention, and their relationship to the processing of context. The model shows how attention can be viewed as the effect that context has on selecting the appropriate pathway for responding. Here, context is provided by the task instructions. Thus, when subjects are presented with conflicting input in two dimensions (e.g., the word GREEN in red ink), they respond to one dimension and not the other, depending upon the *context* in which it appears (i.e., the task: color naming or word reading). If frontal cortex is responsible for maintaining this context, and if schizophrenia involves a disturbance of frontal lobe function, then we should be able to simulate schizophrenic performance in the Stroop task by disturbing processing in the task demand module. More specifically, if frontal lobe dysfunction in schizophrenia is due to a reduction in the activity of its dopaminergic input, then we should be able to simulate this by reducing the gain of units in the task demand module.

Figure 3 shows the results of such a simulation, in which the gain of only the task units was reduced; all other units were unperturbed. This change in the context (task demand) module produced effects similar to those observed for schizophrenics: an increase in overall response time, with a disproportionate increase for color naming interference trials. Thus, the model shows that a lesion restricted to the mechanism for processing context can produce both an overall degradation in performance as well as the expected attentional deficit.

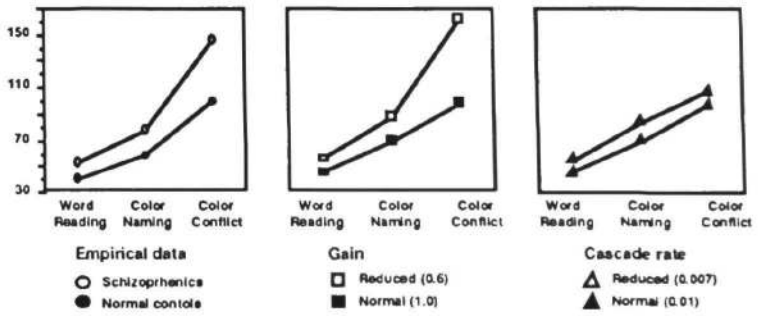


Figure 3

The model also allows us to compare the effects of this specific disturbance to those of a more general disturbance, addressing a common difficulty in schizophrenia research. It is often argued that, in the presence of a general degradation of performance in schizophrenics (e.g., overall slowing of response), it is difficult to know whether degradation in a particular experimental condition is due to a specific deficit or a more generalized one. However, this difficulty arises primarily when the mechanisms for the deficits involved have not been specified. The model provides us with a tool for doing this. Above, we described the mechanism for a specific attentional deficit related to disturbances of dopamine activity in PFC. To compare this to a more generalized deficit, we induced overall slowing in the model by decreasing the rate at which information accumulates for each unit (cascade rate); this was done for *all* units in the model (third panel in Figure 3). A lower cascade rate induced an overall slowing of response, but no disproportionate slowing in the interference condition. In contrast, the specific disturbance in context representation produced both effects. Thus, the context hypothesis provides a better account for the data than at least one type of generalized deficit. We have explored others (e.g., an increase in the response threshold), with similar results.

Simulation of the Continuous Performance Test

The Stroop model shows how contextual information and its attentional effects can be represented in a connectionist model, and how a specific biologically relevant disturbance in this mechanism can explain aspects of schizophrenic performance. One question we might ask is: How general are these findings? Here, we extend the principles applied in the Stroop model to account for performance in the CPT.

As we discussed earlier, schizophrenics show consistent deficits in the CPT. This is particularly true for variants in which a demand is placed on memory for context. For example, in the CPT-double, a target consists of any consecutive re-occurrence of a letter (e.g., a 'B' immediately following a 'B'). Thus, subjects must remember the

previous letter, which provides the necessary context for responding to the subsequent one. Schizophrenics perform poorly in this task. This may be due to an impairment in the processing of context that, like deficits in the Stroop task, might be explained by a reduction of dopaminergic tone in prefrontal cortex. If this is so, then we should be able to simulate schizophrenic deficits in the CPT-double using the same manipulation used to produce deficits in the Stroop task: a reduction of gain in the module responsible for representing and, in this case, maintaining context. To test this, we constructed a network to perform the CPT-double.

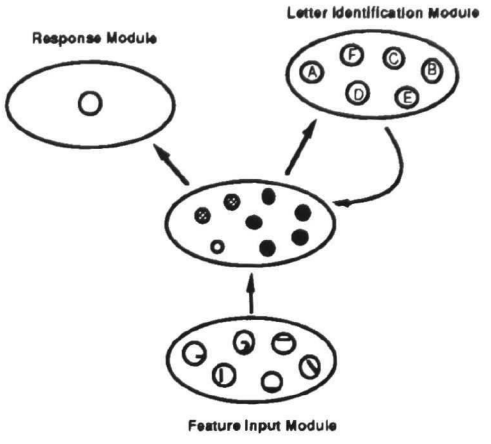


Figure 4

The network consisted of four modules: an input module, an intermediate (associative) module, a letter identification module and a response module (see Figure 4). The input module was used to represent the visual features of individual letters. Stimulus presentation was simulated by activating the input units corresponding to the features of the stimulus letter. The network was trained to associate these input patterns with the corresponding letter units in the letter identification module. In addition, the network was trained to activate the unit in the response module whenever a stimulus letter appeared twice or more in a row. This was made possible by introducing a set of connections from the letter units back to the intermediate units. This allowed the network to store and use information about the previous as well as the current stimulus (see Cohen & Servan-Schreiber, 1989 for a more complete description of training and processing in this model). Note that there is a direct analogy between the role played by the letter units in this model, and the role played by the task demand units in the Stroop model. The representation over the letter units in the CPT model provided the context for disambiguating the response to a particular pattern of input, just as the task demand units did in the Stroop model. In the CPT model, however, context was determined by the previous input, and therefore changed from trial to trial.

Following training, the network was able to perform the CPT-double task perfectly for a set of 26 different stimuli. To simulate the performance of normal subjects — who typically miss on 13% of trials and false alarm on 1% of trials (see Figure 5a) — noise was added to the net input, with the amount adjusted to match the performance of the network with that of human subjects. The results of this simulation appear in Figure 5b (gain = 1.0). Then, to simulate schizophrenic performance, we disturbed processing in the letter module — which was responsible for representing and maintaining context — by decreasing the gain of these units by an amount comparable to the amount used in the Stroop simulation (0.66). The percentage of misses increased to 20%, while false alarms increased slightly to 1.1%. These numbers closely match the results of empirical observations of schizophrenic subjects.

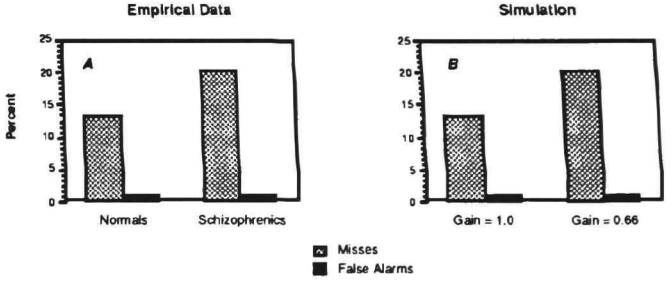


Figure 5

Although some authors have interpreted schizophrenic performance in the CPT in terms of a deficit in sensory processing, our model suggests an alternative hypothesis: Performance deficits are due to a degradation in the memory trace required — as context — for processing the current stimulus. We assume that this memory trace is maintained in prefrontal cortex, and is directly influenced by changes in the dopaminergic supply to this area. This hypothesis is consistent with our account of Stroop performance, and with disturbances of language processing that we turn to next.

Simulation of Context-Dependent Lexical Disambiguation

The language model (Figure 6) incorporates elements of the two previous simulations. The network was similar to the CPT model . It was trained to associate input patterns representing lexical stimuli (e.g., the word PEN) to patterns in two output modules: a response module and a discourse module. Patterns in the response module specified the meaning of the input words (e.g., "writing implement"), while the discourse module represented the topic of the current sequence of inputs (i.e., the meaning of the sentence, rather than the meaning of individual words). As in the CPT model, there were two-way connections between the intermediate module and the context (discourse) module. Thus, once a discourse representation had been activated by an input pattern, it could be used to influence the processing of subsequent stimuli in the semantic module. This provided the mechanism by which context could be used to resolve lexical ambiguity.

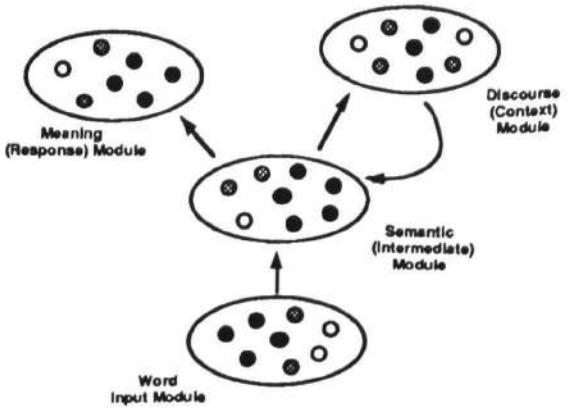


Figure 6

The model was trained to produce an output and discourse representation for 30 different input words, some of which were ambiguous. In the case of ambiguous words, the model was trained to produce the response and discourse patterns related to one meaning (e.g., PEN → "writing implement" and *WRITING*) more than the other (e.g., PEN → "fenced enclosure" and *FARMING*). This asymmetry of training was similar to that of the Stroop model (trained on words more than colors), with a comparable result: when presented with an ambiguous input word, the network preferentially activated the strong (more frequently trained) response and discourse representations. To permit access to the weaker meaning, the network was sometimes presented with an ambiguous word as input along with one of its associated discourse representations (e.g., PEN and *FARMING*), and trained to generate the appropriate response (i.e., "fenced enclosure"). Finally, the network was trained on a set of context words, each of which was related to one meaning an ambiguity; these words (e.g., CHICKEN) were trained to produce their own meaning as the response ("fowl") as well as a discourse representation that was identical to the corresponding meaning of the related ambiguity (*FARMING*). The combined effects of these training procedures was that when an ambiguous word was presented and there was no representation active over the discourse units, the output was a blend of the two meaning of the word, with elements of the more frequently trained (dominant) meaning being more active than the other (subordinate) meaning. However, when a discourse representation was active, the model successfully disambiguated the input and activated only the relevant meaning response.

We tested the model's ability to simulate — in very simple form — the use of context in natural language processing. Most words in English have more than one meaning; therefore, language processing relies on context provided by prior stimuli to disambiguate current ones. In the model, this was achieved by constructing a discourse representation in response to each lexical input, which could then be used as context for processing subsequent stimuli. We tested the model for this ability by first presenting it with a word related to one of the meanings of an ambiguity (e.g., CHICKEN), allowing activation to spread through the network, then presenting the ambiguity (e.g., PEN) and observing the output. Note that, in this case, the model was not directly provided with a discourse representation. Rather, it had to construct this from the first input, and then use it to disambiguate the second. Tested in this way with all context-word/ambiguous-word pairs (e.g., either CHICKEN or PAPER followed by PEN), the model was able to consistently generate the meaning response appropriate for the context.

To simulate performance in our experiment, the model was presented with pairs of context and ambiguous words (representing the clauses used in the experiment) in either order. Following each pair, the network was probed with the ambiguous word, simulating the subjects' process of reminding themselves of the ambiguity, and choosing its meaning. At each time step of processing, a small amount of noise was added to the activation of every unit. The amount of noise was adjusted so that the simulation produced an overall error rate comparable to that of control subjects. The model's response on each trial was considered to be the meaning that was most active over the output units after the probe was presented. To simulate schizophrenic performance, we introduced a disturbance analogous to the one in the CPT model: a reduction in gain of units in the context module. The results of this simulation (shown in Figure 7, along with the results from the empirical study) show a strong resemblance to the empirical data. They demonstrate both significant effects: a) in the low gain mode, the simulation makes about as many more dominant response errors as do the schizophrenic subjects; however, b) as with the human subjects, this only occurred when context came first. (The number of unrelated errors — not shown in Figure 7 — was approximately the same in both the low gain and normal gain mode.) The model provides a clear view of this relationship between dominant response bias and memory. When gain is reduced in the context module, the representation of context is degraded; as a consequence, it is more susceptible to the cumulative effects of noise. If a contextual representation is used quickly, these effects are less, and the representation is sufficient to overcome a dominant response bias. However, if time

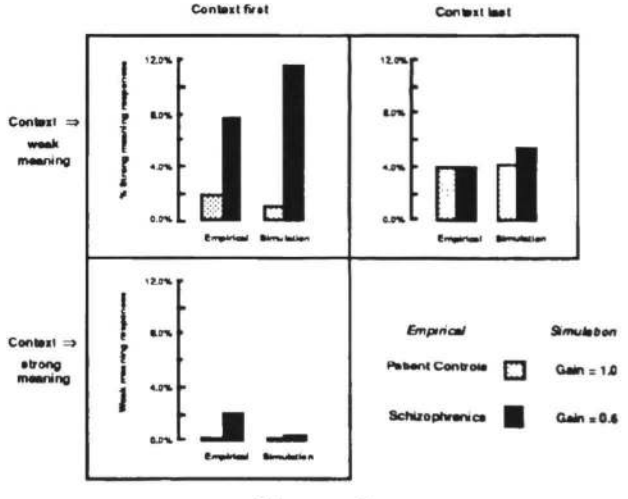


Figure 7

passes (as when context is presented first), the effects of noise accumulate, and the representation is no longer strong enough to mediate the weaker of two competing responses.

Conclusion

The three models we have presented showed how the connectionist framework can be used to link previously unrelated biological and behavioral observations. This was achieved by bringing a common set of mechanisms to bear simultaneously on physiological and psychological phenomena. Specifically, the models served several purposes: a) they simulated quantitative aspects of performance in three previously unrelated behavioral tasks; b) they elucidated the role of processing of context in both the attentional and linguistic tasks; c) they related processing of context to biological processes; and d) they showed how a specfic disturbance at the biological level could account for schizophrenic patterns of performance.

References

Abramczyk RR, Jordan DE & Hegel M (1983). "Reverse" Stroop effect in the performance of schizophrenics *Perceptual and Motor Skills 56*:99-106.

Brozoski TJ, Brown RM, Rosvold HE, & Goldman PS (1979). Cognitive deficit caused by regional depletion of dopamine in prefrontal cortex of Rhesus monkey. *Science 205*:929-931.

Chapman LJ, Chapman JP & Miller GA (1964). A theory of verbal behavior in schizophrenia. In Maher BA (Ed.), *Progress in Experimental Personality Research, Vol 1*. New York: Academic Press.

Chiodo LA & Berger TW (1986). Interactions between dopamine and amino acid-induced excitation and inhibition in the striatum. *Brain Research 375*:198-203.

Cohen JD, Dunbar K & McClelland JL (1990). On the control of automatic processes: A parallel distributed processing model of the Stroop effect. *Psychological Review*

Cohen JD & Servan-Schreiber D (1989). A parallel distributed processing approach to behavior and biology in schizophrenia. Technical Report AIP-100. Department of Psychology, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

Cohen JD, Targ E, Kristoffersen T & Spiegel D (1990). The fabric of thought disorder: Disturbances in the processing of context. In submission; available upon request.

Cohen RM, Semple WE, Gross M, Nordahl TE, DeLisi LE, Holcomb HH, King AC, Morihisa JM & Pickar D (1987). Dysfunction in a prefrontal substrate of sustained attention in schizophrenia. *Life Sciences 40*:2031-2039.

Diamond A (1989), *The development and neural bases of higher cognitive functions*. New York: NY Academy of Science Press

Foote SL, Freedman R & Oliver AP (1975). Effects of putative neurotransmitters on neuronal activity in monkey auditory cortex. *Brain Research 86*:229-242.

Freeman WJ (1979). Nonlinear gain mediating cortical stimulus-response relations. *Biological Cybernetics 33*:243-247.

Fuster JM (1980). *The prefrontal cortex*. Raven Press.

Goldman-Rakic PS (1985). Circuitry of primate prefrontal cortex and regulation of behavior by representational memory. *Handbook of Physiology - The Nervous System 5*:373-417.

Ingvar DH & Franzen G (1974). Abnormalities of cerebral flow distribution in patients with chronic schizophrenia. *Acta Psychiatrica Scandinavia 50*:425-462.

Kornetsky C (1972). The use of a simple test of attention as a measure of drug effects in schizophrenic patients. *Psychopharmacologia (Berl.) 8*:99-106.

Malmo HP (1974). On frontal lobe functions: Psychiatric patient controls. *Cortex 10*:231-237.

Meltzer HY & Stahl SM (1976). The dopamine hypothesis of schizophrenia: a review. *Schizophrenia Bulletin 2*:19-76.

Nuechterlein KH (1984). Information processing and attentional functioning in the developmental course of schizophrenic disorders. *Schizophrenia Bulletin 10*:160-203.

Servan-Schreiber D, Printz H, and Cohen JD (1990) A neural network model of catecholamine effects: Enhancement of signal detection performance is an emergent property of changes in individual unit behavior. in DS Touretzky (ed.)*Advances in Neural Information Processing Systems 2*. San Mateo, CA: Morgan Kauffman.

Wapner S & Krus DM (1960). Effects of lysergic acid diethylamide, and differences between normals and schizophrenics on the Stroop color-word test. *Journal of Neuropsychiatry* 2:76-81.

Weinberger DR & Berman KF (1988). Speculation on the meaning of metabolic hypofrontality in schizophrenia. *Schizophrenia Bulletin 14*:157-168.

Weinberger DR, Berman KF & Zec RF (1986). Physiological dysfunction of dorsolateral prefrontal cortex in schizophrenia: I. Regional cerebral blood flow evidence. *Archives of General Psychiatry 43*:114-125.

A Functional Role for Repression in an Autonomous, Resource-constrained Agent[1]

Michael Fehling
Intelligent Systems
Laboratory
321 Terman Center
Stanford University
Stanford, CA 94305-4025

Bernard Baars
The Wright Institute
2728 Durand Ave.
Berkeley, CA 94704

Charles Fisher
The San Francisco
Psychoanalytic Institute
2420 Sutter
San Francisco, CA 94115

Abstract

We discuss the capabilities required by intelligent "agents" that must carry out their activities amidst the complexities and uncertainties of the real world. We consider important challenges faced by resource-constrained agents who must optimize their goal-directed actions within environmental and internal constraints. Any real agent confronts limits on the quality and amount of input information, knowledge of the future, access to relevant material in memory, availability of alternative strategies for achieving its current goals, etc. We specify a "minimalist" architecture for a resource-constrained agent, based on Global Workspace Theory (Baars, 1988) and on the research of Fehling (Fehling & Breese, 1988). We show how problem-solving and decision-making within such as system adapts to critical resources limitations that confront an agent. These observations provide the basis for our analysis of the functional role of repression in an intelligent agent. We show that active repression of information and actions might be expected to emerge and play a constructive role in our model of an intelligent, resource-constrained agent.

1. Introduction.

Repression involves the purposeful exclusion of information from consciousness. The repressed material is typically assumed to be too painful, shameful, or anxiety-provoking to be tolerable (Freud, 1915). Although observations suggesting repression are widely reported in the psychological clinic, hard experimental evidence for repression remains problematic. Even skeptics, however, agree that some sort of avoidance of painful thoughts is extremely common and demonstrable (Holmes, 1967; Erdelyi, 1985). This paper will not focus on the issue of psychological evidence. Rather, we will ask, could repression, or its functional equivalent, emerge naturally, in any autonomous agent, operating in goal-directed but resource-constrained fashion (Fehling et al., 1989)?

This paper presents a first report of an effort to merge three, rather distinct perspectives on the nature of intelligent agency:

- a functional theoretical perspective that combines a psychological model known as the Global Workspace Hypothesis (Baars, 1988) and a computational model of intelligent problem-solving by resource-constrained agents (Fehling, Altman, & Wilber, 1989),
- a normative theoretical perspective of reasoning adapted from decision-theory and decision analysis, and
- aspects of a Freudian theory of unconscious, psychodynamic processes.

Our discussion of repression illustrates a perspective and methodology that is emerging from our efforts to fuse these distinct theoretical approaches to the nature of intelligent agency.

2. Resource-constrained Agency.

Intelligent problem-solvers and decision makers are *resource-constrained agents*. When acting autonomously in complex, dynamically changing situations that arise in the real world, these agents must cope with the scarcity of resources that may be essential for carrying their tasks. Resource-constrained agents must act to formulate and achieve their objectives without violating constraints imposed by these limited, task-critical resources.

Time is a critical resource. In real-world situations task-performance time is *always* limited. Temporal limitations significantly impact the appropriateness and effectiveness of an agent's actions. An intelligent agent's ability to react to, and interact with, its environment may be constrained by deadlines on the time available to complete those actions. Deadlines require an agent to foreshorten the time spent deliberating about, and carrying out, its actions. Thus, an agent may need to limit the specificity, precision, or quality of its actions in order to meet such real-time constraints. An agent may also need to synchronize its actions with the independent occurrence of environmental events. For example, during diagnostic reasoning an intelligent agent may need to carefully manage the time spent in gathering and interpreting data from various sources in order to observe and interpret intermittently available data (D'Ambrosio et al., 1987).

Information is an equally critical resource. In real-world situations, intelligent agents seldom, if ever, have sufficient information to fully determine the truth of beliefs or unambiguously select the best course of action. Informational incompleteness or uncertainty significantly constrains the effectiveness of an intelligent agent's actions and deliberations. So, an intelligent agent may seek to reduce its uncertainty through *belief management*, including methods to actively gather data and to infer new beliefs from existing ones with sound methods of inference, such as logical or probabilistic deduction. Unfortunately, belief management entails costs as well as benefits (Fehling & Breese, 1988). However beneficial an agent's belief management efforts may be in reducing uncertainty, undertaking these efforts will consume time and other resources. For example, a medical diagnostician may find it quite difficult to decide whether to carry out possibly dangerous exploratory surgery, even though this procedure would return useful information. In addition, by focusing its attention on data-gathering observations or on inferential deliberation, an agent may compromise its ability to achieve other important objectives. Consider a hiker planning the quickest path back to camp before dark. Excessive time in planning could obviate the very goal it is intended to achieve.

An intelligent agent's cognitive capacities represent another set of critical resources. Even the most well-endowed agent is constrained by a finite capacity for storing and retrieving data from memory and rate of inferring new conclusions from previously stored information. While an agent can only indirectly affect the availability of exogenous resources such as time and information, these endogenous, cognitive resources are under more direct control. To cope with its finite capacities, an intelligent agent must judiciously manage the distribution of its cognitive resources among the activities to which they could potentially be applied. Selective attention phenomena may reflect very directly an agent's attempts to cope with the scarcity of its cognitive resources.

Our discussion illustrates how an intelligent agent must make the best use of scarce resources to achieve its most critical objectives. *An agent's effectiveness in reasoning and*

acting depends upon how well that agent can adapt or modify what it might do under ideal circumstances to conform to constraints imposed by the limited availability of critical resources. Theories and computational models of intelligent agency as well as specific problem-solving models must inevitably account for this type of adaptivity.

In general, to achieve this adaptivity, an intelligent agent must seek to maximize the quality of its task performance within the recognized resource constraints. Thus, the agent must employ "meta-level" processes that manage its problem-solving and decision-making activities according to this general adaptivity requirement (Fehling & Breese, 1988). These meta-level processes may

- select the actions or strategies that make the best use of available resouces from previously constructed alternatives
- modify existing actions or strategies, or construct entirely new responses, to simultaneously meet resource constraints and maximize performance quality, or
- some mix of these two basic approaches (Fehling, Sagalowicz, & Joerger, 1986).

Achieving this adaptation is made even more difficult because meta-level processes are themselves problem-solving activity consuming significant resources (Barnett, 1984). The agent must, therefore, bound its efforts to select or construct a plan for accomplishing a resource-constrained task. Meta-level deliberation must leave resources for a suitable task-performance strategy to be enacted (Maes, 1986).

Theorists and experimenters in disciplines such as philosophy, cognitive science, AI, linguistics, and the decision sciences have only recently begun to focus on resource-constrained reasoning. Few investigators have focused on how, or even whether, theories of intelligent reasoning and action might require revision in order to reflect adaptation to limited time, information, cognitive capacity, or other resources. In the psychological literature, most models of intelligent problem-solving, decision-making, etc. make no provision for managing the tradeoffs and compromises required of a resource-constrained agent. There are at least two important counter-examples to this last claim. First, the research and theories on so-called selective attention address issues that are obviously quite similar to those we raise here. In addition, research on limited capacity cognitive mechanisms such as short-term memory or sensory buffer storage focus on important types of endogenous constraints and mechanisms that subjects rely upon in coping with them.

Preliminary investigations of resource-constrained agency suggest that standard conceptions of intelligent reasoning and action may require radical revision (Good, 1983; Fehling et al., 1989). Fehling and his colleagues are conducting in-depth analyses of important cases of resource-constrained agency. Employing the computational modeling perspective of AI and cognitive science, they conclude that a general "architecture" of a resource-constrained agent must include certain features not commonly found in agent architectures such as SOAR (Laird, Newell, & Rosenbloom, 1987), BB1 (Hayes-Roth, 1985), or ACT* (Anderson, 1976). For example, in spite of extensive psychological research on selective attention, none of these architectures provide "interruptability" — the a problem-solving agent's ability to detect the occurrence of critical events, suspend or cancel its on-going activities, and respond to these critical events. Nor do these architectures substantively manage tradeoffs among multiple, independent, and possibly competing tasks. The next section sketches a minimal architecture with some of these features, critically needed by a resource-constrained agent.

Fehling and other AI researchers have argued that that models of specific problem-solving processes of intelligent agents must also be radically revised. Very few problem-solving models provide for resource-constrained, adaptive management of problem-solving activity

itself. Fehling (Fehling & Breese, 1988) has begun formulating an approach to resource constrained problem-solving that incorporates concepts from formal decision theory. This work proposes ways that an agent can decide how and when expend resources to avail itself of information that might significantly change its beliefs or its commitments to future action. Our analysis of the functional role of repression in section 4 incorporates some results of this research. We rely there upon the idea we have just outlined — an intelligent agent will attempt to limit its actions, deliberations, and meta-level deliberations to those elements whose costs are likely to be outweighed by their likelihood of improving future conditions for the agent. This principle applies, in particular, to the way in which information is stored and retained by the agent. To summarize our position

> *Repression would naturally arise in barring from use any information whose costs of use outweigh their potential benefits. In our current architecture, this would specifically imply barring access to information from the GW. Information would be repressed rather than erased if it were sufficiently likely that this cost-benefit tradeoff might change in the future.*

3. A Functional Agent Architecture.

We now describe a "minimalist" architecture for an agent who must optimize its goal-directed actions within environmental and internal constraints. Our analysis of the functional role of repression will depend the assumption that a resource-constrained agent has a limited capacity component, such as a working memory, that is roughly equivalent to conscious access in human beings (Baars, 1988).

We are investigating a *global workspace* (GW) architecture. GW architectures have been explored for about twenty years, as general problem-solving frameworks for a variety of problem-solving tasks. Several psychologists suggest analogies between GW architectures and the human nervous system (e.g. Baars, 1988: Norman & Shallice, 1980; Bower & Cohen, 1982; Reason, 1983). Fig. 1 depicts the five elements of our architecture:

1. A "society" of *Specialists* — These modules each handle a particular sub-task,
2. A *Global Workspace* (GW) — The GW manages input from the agent's sensors and outputs of the agent's effectors. It also broadcasts input information to the society as a whole so that appropriate specialists can be "triggered" to react;
3. A collection of specialized *Sensors* and *Effectors* — Active specialists manage sensors and effector. Thus, a sensor (effector) modality will be operative if its managing specialists are active and non-operative otherwise;
4. A *Dominant Goal Hierarchy* (DGH) – The DGH specifies the current precedence hierarchy of goals and objectives to which the agent is currently commited. The DGH also controls access to the GW. Thus, input relevant to top level goals is given precedence over less significant input;
5. A *Background Information State* (BIS) — The BIS includes a collection of *context hierarchies* and *goal hierarchies*. Each context hierarchy encapsulates an internally consistent collection of information that models some aspects the agent's environment or itself. The goal hierarchies may, under the action of certain specialists, become the DGH.

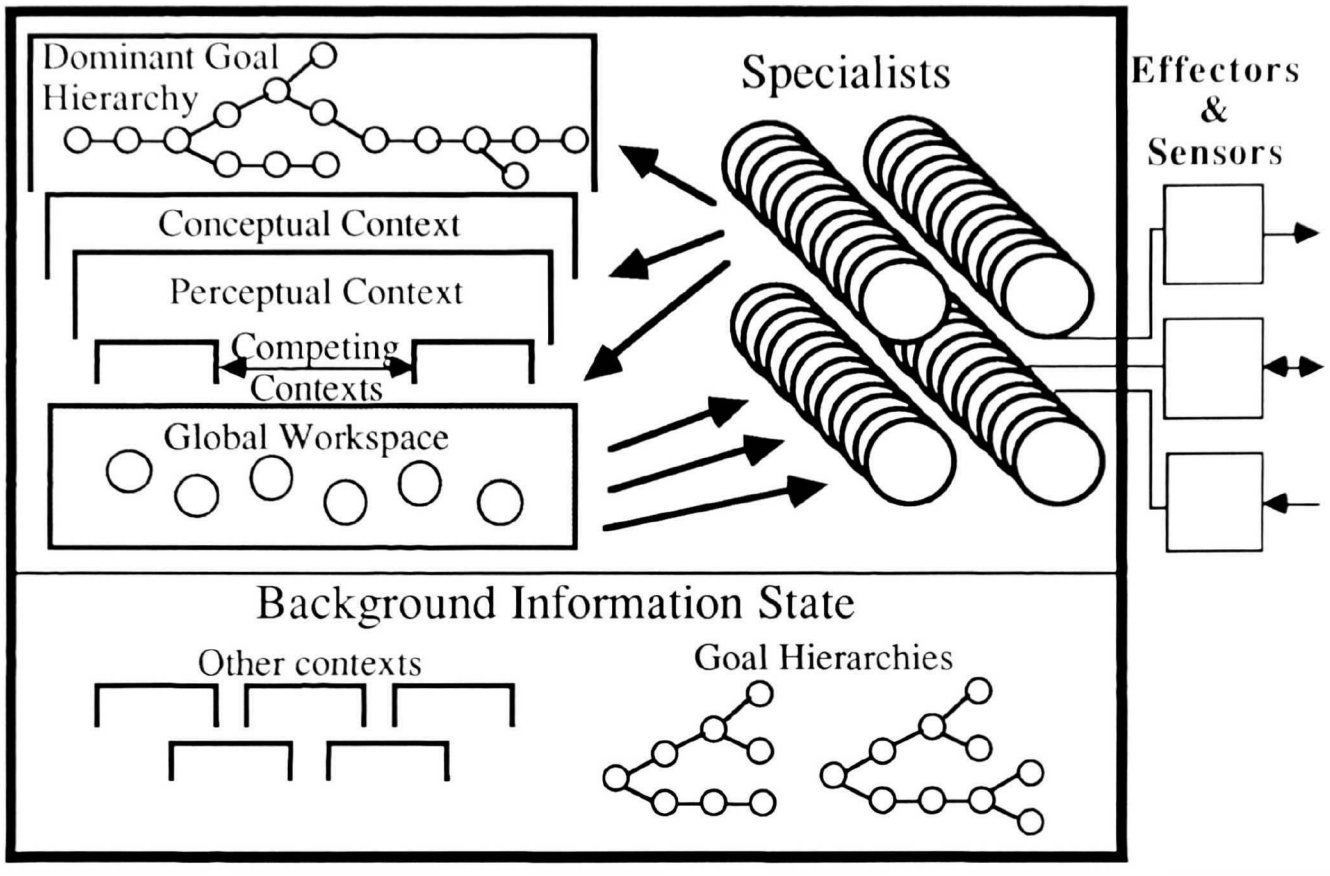


Figure 1 — An Architecture for a Resource-constrained Agent

Section two elucidated how an agent's actions must adapt to contextual constraints. In this light, an additional aspect of our agent architecture is especially important - the ability of the system to detect critical changes in the environment or in itself and react to such critical events. This "interruptability" provides our architecture with the means for coupling its actions to changes in its environment. Fehling, Altman, and Wilber (1989) discuss this aspect of the architecture in detail.

4. Repression as Information Management.

If the Global Workspace is analogous to consciousness or short-term memory (Baars, 1988), the Specialists are comparable to unconscious and isolated processors. Repression is functionally equivalent to purposeful exclusion of some potential GW input from the Global Workspace. Input that for humans provokes pain, shame, guilt, or anxiety has a functional analogue in GW messages that interrupt current GW contents by accessing very high-level goals (such as survival), and perhaps posing problems that demand a radical restructuring of the current Dominant Goal Hierarchy. The question then becomes, *from a pure systems-design perspective, is there a functional role for some mechanism like repression?* We are currently exploring two major possibilities:

1. *The need to avoid goals that have been outgrown, but which cannot be deleted.*

A robotic example. Early developmental stages of an autonomous agent may have created alarm situations (involving the goal of survival) that may no longer be realistic. Let us

suppose that we have used the cognitive architecture of Fig. 1 to implement an autonomous exploratory robot, *Alice, the autonomous agent.* Alice is exploring geological and environmental features of Mars for NASA. Upon her arrival, Alice constructed a base camp, consisting primarily of a large shelter to protect her from dust storms and high levels of solar radiation, and outfitted with solar arrays with which Alice recharge her batteries. When Alice first arrived, she was programmed with sketchy knowledge about the likelihood of danger from dust storms and radiation. In fact, she was programmed with a "cautious" goal hierarchy with higher priority elements requiring Alice to monitor carefully for conditions signalling possible dangers. Alice was also programmed with alternative goal hierarchies containing having high-priority requirements to complete experimental missions. In terms of the functional characteristics implied by Fig. 1, Alice's performance in any situation will be a function of the currently chosen DGH. To protect her from the unknown risks of her environment, Alice's designers provided her with a meta-level "rule" stipulating that the "cautious" goal hierarchy be selected as the default DGH. Operating under this DGH, much of Alice's activity is devoted to assessing possible danger, and, when any evidence of danger is noted, seeking shelter in base camp. Due to her excess caution, Alice found that she would seldom complete a task without many precautionary interruptions. After a long period of residency, Alice found that conditions did not warranted her seeking shelter. At this point Alice switched her goal hierarchy to one that specifies task completion as the highest-priority elements. Under this new DGH she would no longer regularly monitor for danger. She estimates, therefore, that she can devote more time and her own processing resources to completing her experiments. By calculating the expected value of continuing to operate cautiously, but inefficiently, and comparing that to the expected value of operating without precautionary actions, Alice determines that her expected productivity (including her chances of survival) will be higher when using the task-oriented goal hierarchy. This sort of deliberation is known in decision-theoretic terms as a "value-of-information" calculation. Alice has determined that the marginal value of precautionary data-gathering and action is negative in comparison to other modes of operation.

In terms of our architecture, Alice's switch in DGH requires barring data that could activate the cautious goal hierarchy from the GW. Otherwise, this hierarchy could supersede the new DGH. So, information and actions that could select the cautious goal hierarchy must be effectively repressed. Because the decision to repress was based on uncertain information, this goal hierarchy, and the information associated with it, should not be erased, however. Note also that Alice not only precludes the further use of the old goal hierarchy, but *avoids even testing* the relevance of the old goal hierarchy: Such tests would, after all, amount to using the same information-gathering commitments that she has determined not to be cost-effective.

A human example. For human infants, avoiding abandonment is a survival goal, and the infant is extraordinarily vigilant against any hint of rejection. If growing up involves adopting new, more mature, realistic, and self-confident DGHs, superseding the infantile DGH, then interruptions from the infantile DGH must be warded off. But, in ambiguous situations — and practically all social situations are profoundly ambiguous — the tendency to over-interpret input signals in the direction of the infantile DGH may take over again. In this case, *repression is effectively the attempt of the current DGH to guard against even momentary interruptions from the old, and currently non-Dominant Goal Hierarchy.*

But why not simply delete those parts of an outmoded goal hierarchy that have become irrelevant? The dilemma here is to determine when one can confidently delete goals. It is obviously important for an agent to attach probabilities to threats to its well-being. For example, although social rejection may be viewed as a threat to be avoided early in life, perhaps it could be deleted as a threat under evidence that it is irrelevant to well-being.

Indeed, becoming an adult involves falsification of certain childlike dependency goals. However, the threat of abandonment is, in fact, still present. Abandonment may again become a threat under circumstances which create a renewed state of helplessness. Since such circumstances are hard to predict, it would be risky to delete part of a goal hierarchy pertaining to abandonment. This inherent uncertainty may be heightened if tests of the old goal are avoided. Indeed, adult repression often exists for avoidant goals that are never tested and hence never falsified. So, to-be-avoided goals may be given a much higher probability than they actually have. Psychodynamic and cognitive-behavioral researchers have found that many people fear rejection to the point of never testing the reality of rejection, thus also losing the benefits of acceptance and social support (Erdelyi, 1985).

2. *The need to maintain a valued self-concept, i.e., one that is at or near the top of the goal hierarchy.*

A second reason to exclude material from the GW is to maintain a self-concept that ranks the self among the highest values of the hierarchy. Since GW information is widely distributed, it can also be used to revise the goal hierarchy itself. Thus, if we display GW information that shows a high-level goal to be less worthwhile than previously thought, such a goal may be pushed down in the goal hierarchy. Humans are well-known to say things like, "I'll just die from shame if I fail this exam, if I look ridiculous in front of my friends, if I act like a baby," etc. In all these cases, the value of the self is subordinated to a goal. Flexibility in the assignment of values to goals is of obvious importance, for example, in being able to make use of unanticipated opportunities to achieve goals that are not currently on top; or, if a major inconsistency is discovered between two goals, to be able to devalue one of the two goals, so as to achieve at least one. It is important to be able to value a goal nearly as much, or perhaps more than life itself. There is extensive psychological evidence from studies of depression and other disorders that people consciously under-value themselves compared to other goals quite often.

Two contrary tendencies contend. In order to perform major goals, it is useful to broadcast GW message that will place the most important goals high in the goal hierarchy. But, if we place current goals higher than self value, the system may risk self-destruction. *Repression may then protect the global workspace against messages that would serve to posit goals as more valued than the self.* From this point of view, the value of the self-concept must reside at or near the top in the goal-hierarchy, so that it can work to exclude material from the GW that will tend to devalue the self.[2]

5. Conclusions.

We have presented a model of problem-solving by intelligent, *resource constrained agents*. Our *global workspace* architecture appears to meet the minimal requirements for such an agent. We have argued that a resource-constrained agent must bar certain information from its GW, while simultaneously preserving the information in a sequestered manner for use under conditions that might arise in the future. This active exclusion of information from the global workspace corresponds to the idea of repression defined by Freud (1915) as "turning something away, and keeping it at a distance form the conscious." Repression, thus defined, is a necessary aspect of information management in *resource constrained agents* in general and in human beings in particular. While our reasoning differs from that leading to the notion of repression in psychoanalysis, our conclusion is similar. Indeed, contemporary psychoanalytic theory (e.g., Brenner 1982) regards repression as a general

[2]Recall that the DGH has both the function of controlling access to the GW and of maintaining goals for the system as a whole.

mental capacity distinct from a notion of psychopathology. We view repression as a functional concept with profound implications for the behavior of intelligent systems.

We have offered an "argument by design," that seeks to avoid controversies regarding empirical evidence for repression. This approach supports repression's plausibility by (a) proposing plausible designs, or "computational architectures," of intelligent agents, and (b) exploring a design for the role, if any, of repression. The cognitive architecture described in this paper is derived from extensive research by Baars, Fehling, and their colleagues. It is the only one of which we are aware that is explicitly formulated to address the issues of resource-constrained agency. Within this architecture, a mechanism like repression arises naturally and plays an important functional role -- that is, it appears to provide a direct, efficient way for to handle a category of highly likely information management problems.

References.

Anderson, J.R. *Language, Memory, and Thought.* Hillsdale, N.J.: Erlbaum and Associates, 1976.

Baars, B. *A Cognitive Theory of Consciousness.* New York: Cambridge University Press, 1988.

Barnett, J. "How much is control knowledge worth?" *Artificial Intelligence*, 1984.

Bower, G.H. & Cohen, P.R. "Emotional influences in memory and thinking." In M.S. Clark & S.T. Fiske (Eds.), *Attention and Cognition.* Hillsdale, N.J.: Erlbaum & Associates, 1982, pp. 291-331.

Brenner, The Mind in Conflict.

D'Ambrosio, B., Fehling, M.R., Forrest, S., Raulefs, P. & Wilber, B.M.

Erdelyi, M.H. "A new look at the New Look: Perceptual Defense and Vigilance", *Psychological Review*, 81, 1985, pp. 1-25.

Fehling, M.R., Altman, A. & Wilber, B.M. "The HCVM: An Instance of Schemer, an Architecture of Resource-constrained Problem-Solving." In

Fehling, M.R. & Russell, S. *Proceedings of the AAAI 1989 Symposium on Limited Rationality.* American Association of Artificial Intelligence, Menlo Park, CA, March, 1989.

Freud, S. Repression. In James Strachey (Ed.), Standard Edition of the Complete Psychological Works of Sigmund Freud, Vol. XIV, pp. 141-158, London, The Hogarth Press, 1957 (originally published in 1915).

Good, I.J. *Good Thinking: The Foundations of Probability and Its Applications.* Minneapolis: University of Minnesota Press, 1983.

Hayes-Roth, B. "A Blackboard Architecture for Control." *Artificial Intelligence,* 1985, 26(3), pp. 251-321.

Holmes, D. "Closure in a gapped circle figure." *American Journal of Psychology*, 80, 1967, pp. 614-618.

Laird, J.E., Newell, A., & Rosenbloom, P.S. "SOAR: An Architecture for General Intelligence." *Artificial Intelligence,* 1987, 33(1).

Maes, P. *Proceedings of the Workshop on Meta-level Architectures and Reflection,* Sardinia, 1986.

Norman, D.A. & Shallice, T. "Attention and action: Willed and automatic control of behavior." Unpublished manuscript. Center for Information Processing, UCSD, La Jolla, CA, 1980.

Reason, J. "Absent-mindedness and cognitive control." In J. Harris & P. Morris (Eds.) *Everyday Actions, Memory, and Absent-mindedness.* New York: Academic Press, 1983.

Simon, H. *The Sciences of the Artificial.* 2nd Ed. M.I.T. Press. Cambridge, Mass, 1981.

A Goal-Based Model Of Interpersonal Relationships

Stephen Slade
Computer Science Department
Yale University
P.O. Box 2158, Yale Station
New Haven, CT 06520-2158
slade@cs.yale.edu

Abstract

Interpersonal relationships are a pervasive dimension of human behavior and decision making. Actors make choices based both on personal goals, and on goals derived from interpersonal relationships. We present a goal-based model of decision making that combines the motives of the actor with agendas adopted through relationships. A unifying feature of the model is the use of importance as a means of ranking both goals and relationships. We describe a computer simulation of the model in the domain of Congressional roll-call voting.

1 Introduction

In making a decision, a person must often consider the goals of others. An actor's relationships with other people will influence such decisions. In this paper, we discuss a specific model of interpersonal relationships. This model serves as the basis for a computer simulation in the domain of Congressional roll-call voting. VOTE uses knowledge representations of selected members of the House of Representatives, their voting records, ideologies, and relationships with constituency groups to derive and justify voting decisions. Here is a brief example.

```
> (vote 'udall 'plant-closing)
* Member:        Morris K. Udall
* Bill:          Plant Closing Layoff Notice Bill
* Requires employers of more than 100 workers to give 60 days
* notice of a plant closing or layoff lasting more than 6 months.
```
— intermediate output deleted —

```
* English rationale:

Morris K. Udall votes for bill S-2527, Plant Closing Layoff
Notice Bill.  He believes the adverse effects of this bill are
far outweighed by other issues.  He readily endorses the
proposal of requiring 60 days notice prior to closing a factory.
Udall is strongly in favor of the legitimate rights of decent
working people.  Udall strongly supports the principle of
fairness in society.  He believes in the principle of government
regulations in the national interest.  At the same time, he
realizes that the country as a whole approves of the principle
of free enterprise and capitalism.
```

In arriving at a decision, VOTE infers the implications of a particular bill for the relevant constituency groups. VOTE has a natural language generation capability for expressing the justification of the decision. In this paper, we first discuss the model of goal-based decisions and interpersonal relations that underlies the VOTE program. We then examine the representation of relationships in VOTE in the context of the Udall/Plant-closing example.

2 Goals, Importance, and Relationships

In making a decision, an actor often faces trade-offs. Decisions may involve conflicts among goals and resources. A realistic model of decision making and planning must account for a multitude of goals. However, all goals are not created equal. We use *importance* as a measure for ranking goals.

> **Principle of Importance.** The importance of a goal is proportional to the resources which the actor is willing to expend in pursuit of that goal.

That is, the relative importance of goals is determined when goals compete for the same resource. The more important goal is allocated the resource, all other things being equal. Importance here is equivalent to Wilensky's *value* [Wil83].

Given two alternatives, an actor contemplates which choice is *better*, that is, which choice achieves the more important set of goals. We should note that the definition of *better* is not always an easy proposition. The straightforward methods of decision analysis [Rai68] of assigning weights and probabilities to outcomes often finesse issues of cognitive significance, such as memory and attention constraints.

People are interdependent. Many of a person's common goals require the help of another person. Given that individuals differ in goals, resources, experience, and other areas, it is natural that the relationships between individuals will be idiosyncratic as well. We argue that interpersonal relationships will reflect the underlying idiosyncratic goal structures of the individuals.

However, even though interpersonal behavior will exhibit wide variation as a function of the individuals involved, we can derive a set of standard dimensions, such as the following.

- *Goal achievement.* We commonly view relationships as examples of cooperative behavior. That is, we get others to achieve goals for us, and we in turn may satisfy their goals, either directly or indirectly. We vote for a candidate whose record reflects our beliefs.

- *Goal development.* Some relationships cause us to develop new goals for ourselves, rather than merely satisfy the goals of another actor. A political leader may inspire people to adopt a new cause.

- *Importance.* We wish to ascribe importance to a relationship in a manner uniform with our use of importance to describe goals and resources. That is, the more important a relationship, the more likely a person will be to allocate resources for goals affected by that relationship. The US is more likely to send aid to Israel than to Sri Lanka.

- *Symmetry.* Relationships are bilateral. Two people are involved. However, each party may have a different view of the relationship, as well as a different view of the other party's role in the relationship. For example, a Congressman may not treat his relationship with the NRA with the same importance that the NRA does. Such a relationship would be asymmetric.

We wish to distinguish between interpersonal relationships and interpersonal role themes [SA77, Dye82]. A relationship is binary, that is, between two parties. A role theme is n-ary, that is, a collection of relationships. For example, the waitress role theme has numerous relationships: with the customer, with the chef, with the maitre d', with the bus boy, with the manager, with other waitresses, with the customers of other waitresses, and so forth. Associated with each of these relationships are interleaving goals. That is, there are two actors who engage in plans that affect each other's goals. In effect, each actor has *adopted* some of the perceived goals of the other actor.

> **Principle of Interpersonal Goals.** Adopted goals are processed uniformly as individual goals, with a priority determined by the importance of the relationship.

Thus, the importance of the relationship determines what relative importance will be assigned to the adopted goals. The principle of importance applies to adopted goals, meaning that a person will expend resources in pursuit of an adopted goal in proportion to the importance of that adopted goal.

The principle of interpersonal goals encompasses various goal-based phenomena related to importance, such as the following:

- *Resource allocation.* An actor will be willing to expend more resources on an adopted goal if the affected relationship is of greater importance. You are more likely to help a friend than a stranger.

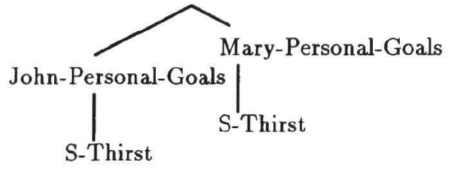


Figure 1: John-Mary Relationship: Mary Regnant

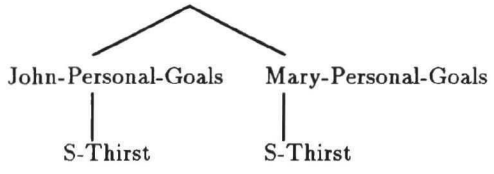


Figure 2: John-Mary Relationship: Egalitarian

- *Cognitive resources: attention.* You would expect to spend more time thinking about the interests or problems of a close friend than those of an acquaintance.
- *Cognitive resources: memory.* It should be easier to remember information about a friend than about a casual classmate.
- *Affect.* You will be more likely to experience an emotion relating to an adopted goal if the relationship is of greater importance. Also, the intensity of the emotion will reflect the importance of the relationship. You are more upset at the death of a parent than that of a neighbor.

Most of our examples will focus on the phenomenon of resource allocation, however, we claim that the cognitive phenomena are similarly extended to this goal-based model of interpersonal relationships.

Consider a simple example. John, who is thirsty, has a girlfriend, Mary, who is also thirsty. They both want some milk. If there is only enough milk for one person, John may give all the milk to Mary, indicating that he has adopted her goal of satisfying her thirst, and decided that his relationship places her needs above his. This situation is depicted in figure 1, which indicates that John has adopted Mary's goals at a level higher than that of his own personal goals. They have equal desires to satisfy thirst, but it is important to John to satisfy Mary's goals in general. Therefore, he will sacrifice his own desires.

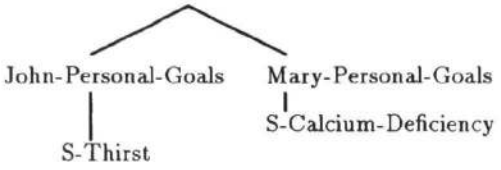


Figure 3: John-Mary Relationship: Unequal needs

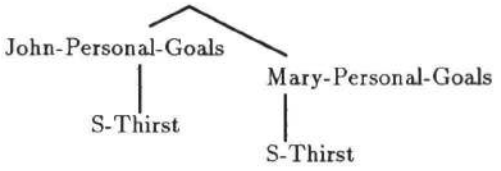


Figure 4: John-Mary Relationship: John Regnant

Alternatively, John may have an egalitarian view of their relationship, suggesting that they share the milk. This situation is diagrammed in figure 2.

However, if Mary has an acute calcium deficiency, making her need to drink milk more pressing than John's, John would give her the milk. Figure 3 portrays this state of affairs. The relationship is egalitarian, but Mary has a greater need.

In another scenario, John may have just had an argument with Mary, making him lower the importance factor of their relationship; so while John might be willing to expend resources in achieving high-priority adopted goals, such as in saving Mary's life, he is not going to let his own interests take a backseat, and he drinks all the milk himself. Figure 4 illustrates this situation.

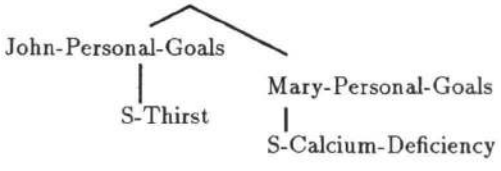


Figure 5: John-Mary Relationship: John Callous

Finally, figure 5 illustrates an extreme situation in which John is simply selfish. Here John places his own less important needs over the more critical needs of Mary.

This model of goal adoption suggests a hierarchy of relationships.

High-priority: Spouse, Children
Self
Parents, Siblings, Close Friends
Colleagues, Partners
Classmates, Neighbors
Low-priority: Strangers

This ordering is merely an example. It indicates someone who cares more for his children than for his parents. It also suggests that the person will put the well-being of his wife and children ahead of his own. The hierarchy of relationships is idiosyncratic and may vary among people and cultures, and even within the same individual at different times in life.

3 Congressional Constituencies

We now turn to the application of the model of interpersonal relationships to a specific computer program, VOTE.

We first should note that Carbonell used goal hierarchies to model political ideologies in adversarial relationships [Car79]. His POLITICS program focused on counterplanning — taking measures to keep your opponent from achieving his goals while preventing him from blocking your own plans. The representation offered here is compatible with Carbonell's model. However, the focus of the VOTE program is on cooperative behavior, rather than counterplanning. Outside the domain of foreign policy, it is usually more important to help your friends than to thwart your enemies.

VOTE, written in T [Sla87], comprises over 12,000 lines of code, and over 8,000 lines of data. The VOTE program relies on a set of interrelated databases, including issues (over 200 currently in the database), constituency groups (150), bills (41), members (68), and decision strategies (20). We note that many decision strategies are required since the explanation of the decision depends on the strategy. It is not enough to use one simple strategy of summing the weights of the conflicting issues and relationships.

In VOTE, we use the names and records of real members of Congress, and attribute relationships and issue stances to them. The coding decisions were made by an expert informant with experience as a political consultant and White House staff member. By way of disclaimer, we should state that these data are merely illustrative, and are not meant to represent the beliefs of actual members of Congress. Thus, we make no claims for the accuracy of these coding decisions, nor of the voting behavior or explanations exhibited by the program. The purpose of VOTE is not to predict individual voting decisions, but rather to

demonstrate the feasibility of a particular model of interpersonal relationships and decision making.

Stances and **Relationships** provide mappings for issues and groups, respectively. Associated with a particular instance will be a level of importance from A (high) to D (low), and a polarity (pro or con).

For example, VOTE has the following representation for the relationships for Representative Udall.

```
Relations:      ((PRO B MEMBER:MEMBER.1025 ADA)
                 (PRO A MEMBER:MEMBER.1025 ACLU)
                 (PRO B MEMBER:MEMBER.1025 COPE)
                 (PRO C MEMBER:MEMBER.1025 CFA)
                 (PRO C MEMBER:MEMBER.1025 LCV)
                 (CON B MEMBER:MEMBER.1025 ACU)
                 (CON B MEMBER:MEMBER.1025 NTU)
                 (CON B MEMBER:MEMBER.1025 NSI)
                 (CON B MEMBER:MEMBER.1025 COC)
                 (CON B MEMBER:MEMBER.1025 CEI)
                 (PRO C MEMBER:MEMBER.1025 DEMOCRATS)
                 (CON C MEMBER:MEMBER.1025 REPUBLICANS)
                 (PRO C MEMBER:MEMBER.1025 COUNTRY))
```

VOTE can express these relationships in English as follows.

```
Morris K. Udall unwaveringly endorses the ACLU's strong defense
of the Bill of Rights.  He feels strongly in favor of the
progressive agenda of COPE, as well as the ADA as a proponent of
basic American values.  He is strongly against the narrow special
interest of the Chamber of Commerce, the CEI business special
interest lobby, the ACU's right-wing reactionism, the National
Taxpayers Union, as well as the NSI as an example of the radical
right.  Udall opposes Republicans.  Udall approves of the
Consumer Federation of America, the League of Conservation
Voters, the country, and members of the Democratic party.
```

In considering the Plant Closing bill given in the opening example, VOTE arrived the following set of stances in favor of the decision (`Reason:`), and one stance in opposition to the decision (`Downside:`).

```
Reason:         (((PRO B GROUP:COPE PLANT-CLOSING)
                  (PRO B GROUP:DEMOCRATS PLANT-CLOSING))
                 ((PRO B GROUP:COPE LABOR)
                  (PRO B GROUP:ADA LABOR)
                  (PRO B GROUP:DEMOCRATS LABOR))
                 ((PRO B GROUP:COUNTRY FAIRNESS))
                 ((CON B GROUP:CFA DEREGULATION)))

Downside:        ((PRO C GROUP:COUNTRY FREE-ENTERPRISE))
```

The key issues supporting this bill were plant-closing (the issue itself, as opposed to the particular bill), labor rights, fairness, and opposition to deregulation. There were proponents for each of these issues: COPE (the Committee On Political Education of the AFL-CIO), the Democratic Party, Americans for Democratic Action, the normative views of the country, and the Consumer Federation of America. Each of these groups had a positive relationship with the Congressman, and the resulting decision reflects their interests and issue agendas.

The downside stance reflects the normative view in support of free-enterprise. While the plant-closing bill involved several other issues not given here, none of them were of interest to this Congressman or his constituents.

4 Conclusion

We have proposed a goal-based model of interpersonal relationships. By allowing adopted interpersonal goals to be processed uniformly with personal goals, the model affords robustness and flexibility. Importance provides a means of ranking both goals and relationships. The VOTE program demonstrates the feasibility of the model in a realistic domain.

References

[Car79] J. Carbonell. *Subjective Understanding: Computer Models of Belief Systems.* PhD thesis, Yale University, 1979. Technical Report 150.

[Dye82] M. Dyer. *IN-DEPTH UNDERSTANDING: A Computer Model of Integrated Processing For Narrative Comprehension.* PhD thesis, Yale University, 1982. Technical Report 219.

[Rai68] H. Raiffa. *Decision Analysis: Introductory Lectures on Choices under Uncertainty.* Addison-Wesley, 1968.

[SA77] R.C. Schank and R. Abelson. *Scripts, Plans, Goals and Understanding.* Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1977.

[Sla87] S. Slade. *The T Programming Language: A Dialect of LISP.* Prentice-Hall, Englewood Cliffs, NJ, 1987.

[Wil83] R. Wilensky. *Planning and Understanding.* Addison-Wesley, Reading, Mass, 1983.

Volition and Advice:
Suggesting Strategies for Fixing Problems In Social Situations *

Eric A. Domeshek
Institute for the Learning Sciences
Northwestern University
Evanston, Illinois 60201

Abstract

Just as an abstract causal analysis of a plan's faults can suggest repair strategies that will eliminate those faults [6], so too, an abstract causal account of how a problem arises in a social situation can suggest relevant advice to correct the problem. In the social world, most problems arise as results of agents' actions; the best way to fix such problems is to modify the behavior that produces the problem. The vocabulary of *volition* developed in this paper is proposed as an abstract level of motivational analysis useful for discriminating among strategies for changing behavior.

Volitional analysis focuses on the agents involved in an action. In addition to the *actor*, there is often a *motivator* agent who influences the actor and sometimes a *third-party* agent used as a tool by the motivator. If any of these agents can be swayed, the problematic action may be avoided. By identifying these agents and classifying the influences working on them, volitional analysis can suggest relevant modifications. The influences most often depend on the social context that links agents and establishes goal-generating themes. Behavior, however, is not always directly goal-governed, and volitional analysis recognizes these exceptional cases as well.

1 Problems in the Social Domain

Consider a situation where a man is seen in the company of a woman; he has a wife, but this isn't her. Imagine his wife is the one who sees him with this other woman. If she considers this a problem, what should she do? The answer of course depends on *why* her husband was in the company of this other woman. Her response depends on "*why*" in at least two senses: the first is whether the reason for the observed action, and therefore its meaning, actually signals any threat to her marriage; the second is that in order to change the behavior, her response should attack its causes. This paper is concerned with the second issue: *the choice of behavior modification strategy.* A reasonable response would be quite different in each of the following circumstances:

1. He was in the middle of one of a long series of secret trysts;
2. He was far from home, lonely, and this woman caught his eye;
3. A friend asked him to keep his sister company;
4. His boss assigned him to entertain this client;
5. He was just holding the door for a random passerby.

Differences between these situations, range from the underlying goals, through the specific actions implied, to the likely effects on the marriage. When the task is to eliminate a problem, however, we can focus on the causes of that problem. To change problematic behavior in the social domain, we can focus on why the actor exhibited the behavior and try to alter that particular causal chain. *Motivational analysis* — the construction of causal explanations for agents' behavior by appeal to goals, plans and other intentional constructs — is a complex and much-studied problem, although within AI, interest seems to have been restricted to the Natural Language Processing community [9, 1, 13, 5, 10].

This paper proposes and justifies a new vocabulary for summarizing complex motivations: *volitional analysis.* The point of this paper is to argue that the vocabulary of volition is particularly useful for discriminating among possible strategies for modifying behavior. The notion of functionally justifying a representational vocabulary by arguing for its fit to some task, (here, counter-planning in the social domain), is discussed in the next section.

This analysis of volition derives from work on the ABBY case-based lovelorn advising system [4]. ABBY's case library is composed of fixed advice packets; when an input problem situation is described in a way that matches the label on a piece of advice, ABBY retrieves the advice and offers it to the user. An extension to the system, currently

*This research was supported in part by the Air Force Office of Scientific Research (AFOSR). The Institute for the Learning Sciences was established in 1989 with the support of Andersen Consulting, part of the Arthur Andersen Worldwide Organization.

being implemented, will retrieve stored advice for inputs whose descriptions include assessments of volition. Sensitivity to volition will help ensure that ABBY's chosen advice will reflect relevant behavior modification strategies.

2 Plan Modification in the Social Domain

Planning systems in AI have historically focused on synthesizing complete and correct plans for specific tasks, building these plans out of simple actions and discarding the results after execution[2]. A newer paradigm — case-based planning [8, 7, 6, 3, 12] — suggests that, when possible, plans are constructed starting from complete solutions to old problems. An important step in case-based planning is *modifying* the old plan to fit the new situation.

To fix a proposed plan, one must be able to characterize what is wrong with it in a way that suggests useful repairs. Hammond [6] has pointed out the importance of understanding the *causal mechanisms* that lead to problems in plans. His CHEF system uses causal analysis to produce abstract descriptions of problem mechanisms which serve as indices to clusters of repair strategies. He demonstrated a class of abstract problem descriptions, derivable from detected problems, and useful for suggesting relevant repairs.

The insight that causal analysis of problem situations can suggest relevant repair strategies applies just as well to *social situations* as to problems centered on physical causality: if we know the social mechanisms underlying a problem, then we can focus on changing aspects of the situation that lead to that problem. Seifert, for example [11], capitalizes on the structure of *mutual goals*, which commonly arise in social situations, to suggest planning strategies.

The sorts of *advice* people offer one another about their everyday social lives can often be viewed as strategies for repairing faulty plans. In giving advice, then, our choice of advice should benefit from sensitivity to causal analysis. In the social world, problems derive from the effects of agents' actions. An especially relevant sort of causality is the mechanisms that determine why agents do what they do. The modification strategies suggested by volitional analysis aim to fix problems by modifying behavior, rather than by changing circumstances to make the same old behavior yield different (unproblematic) results.

Volition analysis is not a substitute for full motivational analysis; the modification strategies it licenses are not substitutes for detailed planning. Volition is a summary vocabulary designed to highlight important behavioral influences and thus suggest ways to effect changes. This paper does not address the difficult issue of how to *generate* these volitional descriptions of actors' relationships to their actions, nor does it demonstrate how to apply the strategies suggested by volitional analysis to produce specific modifications and finished plans. The former is beyond the scope of this paper; the latter is beyond the scope of the ABBY project.

3 Social Causality

Traditionally, in both Naive Psychology and NLP research, we explain individuals' actions by appeal to *goals*. If someone eats, it is likely because they were hungry. More sophisticated analyses recognize the importance of *goal relationships* [13] and posit still higher motivational entities called *themes* [9]. A theme is a relatively persistent property of an agent that functions as a goal-generator; many themes follow from relationships between agents. We explain a choice to *go out to dinner* by a conjunction of the desires to eat and to socialize (a positive goal relationship). We recognize a choice of an *expensive romantic restaurant* as deriving from the specific nature of a romantic relationship, (an interpersonal theme).

If you want to change this behavior, either because it is problematic for the planner, (requiring plan-modification), or because it is problematic for a bystander, (requiring counter-planning aimed at getting the planner to modify his plan), then knowledge of the underlying goals offers some leverage. In the simplest case, acknowledging the operative goals but pointing out another mechanism for achieving them may be an effective modification strategy. More interestingly, pointing out *other* interacting goals may suggest that the action oughtn't be performed, that the original goals ought to be pursued in some other way, or that there is some better method that will achieve still more goals. Going out to a fancy restaurant may take a big bite out of savings being accumulated to finance a vacation; perhaps, thinking of vacations, you've been wanting to get out of the city; maybe a romantic picnic in the countryside would do better.

In a social context, personal goals are not the only mechanisms that cause agents to act. The romantic interpersonal theme introduced specific interpersonal goals affecting the choice of plan, (dinner together at a romantic restaurant). Alternately we could view this as a goal of the group entity — the "relationship" — which is adopted by an individual member of the group. This sort of analysis is clearer in the case of larger groups like families: the Jones family decides to have a reunion, and many of the individual members figure out how to get to California over Christmas.

Viewing the *group* as an agent with its own goals, the transmission of goals from groups to individuals can be viewed as a subclass of the general phenomena identified by Schank and Abelson [9] as *agency*: getting someone to do something for you. They

proposed a **D-agency** goal and accompanying **persuade** plans specifying how one agent can influence another to take some action.

Among the standard methods of persuading is **invoke-theme**. **Invoke-theme** might engage a personal theme, as when a charity plays on someone's self-image as a good liberal. Alternately, the theme in question might be the more specific relationship between the solicitor and the target, as when a college asks its alumni for support based on their teary attachment to the old alma mater. The Jones family example hinged on such group membership.

Persuades that establish agency are often important links in the causality accounting for why agents do what they do in social situations. The **persuade** plans presented in [9] included: **ask**, **invoke-theme**, **inform-reason**, **bargain-object**, **bargain-favor** and **threaten**. We incorporate this fragment of goal/plan motivational analysis into the vocabulary of volition because it indicates when there is a second agent involved in causing an action and because the different forms of persuasion are susceptible to different forms of attack.

4 Types of Volition

In designing a vocabulary for volition, we seek to ensure that it captures distinctions that matter when choosing behavior modifying strategies. We can classify volitions along several dimensions:

- **Source:** The *source* of the impetus to act;
- **Influence:** How the **actor** was *influenced* to act;
- **Choice:** The **actor's** degree of *choice*.

4.1 Source of Impetus

The basic question here is: did the initial impetus to perform the action come from the **actor** (most often in response to one of his own goals), or did it come from someone else? Again, the point is to identify the agents responsible for the action so we can choose strategies to change it. In the first case we consider the action to have been performed under **internal** volition; in the latter, under **external** volition.[1]

In the case of *internal* volition there are several other questions to ask. When the law seeks to assess blameworthiness it asks whether the action was **premeditated** or **spontaneous**. If premeditated, we can ask whether the particular action was thoughtfully **chosen** from among alternatives or whether it was simply the **default** option, adopted without thought or without knowledge of alternatives. If spontaneous, we recognize several sub-classes. Some actions are done for **emotional** reasons that have little to do with rational goal-pursuit; again, the legal world offers a similar distinction in recognizing "crimes of passion." Many actions are thoroughly scripted. It is so **conventional** a part of the normal morning routine to eat breakfast that you need not really think about it. Eating cold cereal for breakfast may be a personally **habitual** routine; again, no thought or decision is required, so you may find yourself eating cereal even on mornings when you don't really want to.

For the most part, agents do things in response to goals — their own or those of others. Sometimes, however, agents are involved in actions for reasons that have nothing at all to do with goals. The prototypical case is **uncontrolled** actions such as sneezing or falling down stairs. Another odd case is when agents do actions **unknowingly**, in the sense of not realizing alternate interpretations of the action; sitting down in the presence of a king may constitute "lese majesty." Finally, agents may *appear* to have done something, but actually have not: they may be **uninvolved** and merely implicated by circumstances. All of these odd cases indicate actions, or construals of actions, that are essentially *non-volitional*.

4.2 Mechanisms of Influence

External volitions are distinguished by the presence of some other agent: the *motivator*. In the case of external volition, the central questions distinguishing among different cases center on how the **motivator** manages to influence the behavior of the **actor**. The persuasion plans mentioned earlier are one set of mechanisms, each suggesting different behavior modification strategies. Consider the differences between changing the behavior of an **actor** who has merely been **asked** to do something, versus one who has been **inspired** or **convinced**. It takes different countermeasures to overcome these varying degrees and sources of commitment to an action.

Many of the more effective plans for persuasion depend on the use of *inducements*, which are defined in terms of the **actor's** goals. As used here inducements may be actual or future, positive or negative; thus they include promises and rewards, threats and punishments. For example, use of the **invoke-theme** plan plays on the notion of **obligation**, and introduces a whole raft of implicit inducements; themes generally subsume many goals, so invoking the theme serves to remind the **actor** of the benefits he can expect from complying, (and thereby maintaining the theme), and also of the loss he can expect if, in refusing, he drives the **motivator** to disrupt the theme. Actions **bought** with money, services or material goods offer the clearest case of inducements. Actions performed in response to arbitrary **threats** illustrate the effectiveness of negative

[1] This paper uses **boldface** type for representational vocabulary items and for the several agent roles identified in volitional analysis (**actor**, **motivator** and **third-party**).

inducements.

Several other special classes of external volition are worth distinguishing. In most sorts of external volition, the **actor** ends up performing the action because another agent manages to tie it to some goal the **actor** cares about. There are however the external analogs of **uncontrolled**, **unknowning** and **uninvolved** actions defined in the previous section. An **actor** can be **compelled** to do an action — he can be physically manipulated and thereby forced to do, or not to do, almost anything. An **actor** can be intentionally **misled** as to what he is doing; another agent can tell him he is invited to a party when he will actually be crashing it. An **actor** can be **framed** — another agent can intentionally arrange things to appear as though the agent performed some action though he has not.

A final broad class of external volitions are best thought of as **third-party** volitions. These introduce a third agent into the causation of an action. The **third-party** may be recruited as a **surrogate motivator**, or the **motivator** may involve the **third-party** either to **administer** or to **receive** inducements. Examples of **third-party** as surrogate include getting someone influential to make **appeal** to the **actor**, or invoking external authority to **police**, and thus compel, action. An example of getting a **third-party** to administer inducements is **blackmail**: the **motivator** threatens to do something that will cause another agent to make problems for the **actor**. An example of **third-party** as recipient is a **hostage** situation: the **motivator** threatens to punish the **third-party**.

To clarify the various roles in these situations we need a linguistic distinction which does not exist in common usage: an action that is motivated externally will be the result of the someone *inciting* the **actor**, often by setting up an *inducement*. The incitement is the communication intended to influence the **actor**. The inducement is an effect on some goal intended to give teeth to the incitement. The point of this distinction is that different agents can deliver the incitement and the inducement. In the case of **blackmail**, for instance, the prototypical situation involves a blackmailer inciting an **actor** to do something by threatening to reveal some fact. But revealing a fact is not an inducement; it is the role of some **third-party** to respond to the revealed fact and actually deliver the negative consequences implied in the original threat. This **third-party** is effectively delivering the inducement.

4.3 Degree of Choice

Degree of choice is an issue when it comes to assessing credit or blame for an action. This way of classifying volitions works with the internal/external distinction to focus attention on particular agents, and with the varieties of influence to mitigate strategies that rely on negative ways of changing behavior. As

Internal	External	Third-Party
Premeditated	Persuaded	Surrogate
Chosen	Asked	Appeal
Defaulted	Convinced	Police
Spontaneous	Inspired	Administer
Emotional	Obliged	Blackmail
Conventional	Bought	Receive
Habitual	Threatened	Hostage
Uncontrolled	Compelled	
Unknowing	Misled	
Uninvolved	Framed	

Figure 1: The Vocabulary of Volition

in the case of legal distinctions cited earlier: it is considered less justified to execute someone for an action they didn't intend to do, or had no choice about doing.

The vocabulary of volition is outlined in Figure 1. It is intended to be systematic and representative, not necessarily exhaustive. Having introduced all these categories, we must now show how they capture differences in the applicability of strategies for plan modification. In using causal analysis to discriminate among plan repair strategies CHEF capitalizes on the insight that knowledge of how a state was caused suggests ways to disrupt the causal chain and eliminate the effect. In ABBY, the same basic principle translates as: *knowing how an agent came to do an action allows us to work out ways to get him not to do it.*

In both cases, we need to abstract from the details of any particular causal chain, fitting the specific instance to a set of categories that suggest relevant modification strategies. The volition types listed here are those abstract categories. Identifying the degree of choice an **actor** had and the internal goals or external influences that prompted the action tell us where our points of leverage may be in getting the **actor** to stop doing something, do it differently, undo what they've done, or not do it again.

5 Behavior Modification Strategies

The mapping from volition to behavior modification strategies is based on the causal model underlying each volition. This section presents a series of tables sketching the causality underlying each type of volition and showing how each causally significant fact suggests ways to change the behavior. The left column contains the causally relevant facts; the right contains strategies that seek to change those facts (and thus the resulting behavior).

For example, Figure 2 illustrates the differences along the "sources" dimension. Distinguishing internal from external volition and noticing involvement of other agents focuses attention on those agents who help cause problematic actions: the *actor*, the *motivator*, and the *third-party*. Only when you have assessed an **actor**'s volition with respect to a prob-

Model	Strategy
Internal	
Actor does Act	Change **Actor's** behavior
External	
Actor does Act	Change **Actor's** behavior
Motivator influences **Actor**	Change **Motivator's** influence on **Actor**
Third-Party	
Actor does Act	Change **Actor's** behavior
Motivator affects **Third-Party**	Change **Motivator's** effect on **Third-Party**
Third-Party influences **Actor's** behavior	Change **Third-Party's** influence on **Actor**

Figure 2: The Three Broad Classes of Volition

Model	Strategy
Premeditated	
Actor has Goal	Get **Actor** to drop goal
Actor does Act as part of Plan	Get **Actor** to do different Act for Plan
Chosen	
Actor considers possible Plans for Goal	Introduce new Plans into consideration Remove Plan from consideration
Actor adopts Plan for Goal	Get **Actor** to adopt other plan
Defaulted	
Actor retrieves normal Plan for Goal	Tell **Actor** about new Plan
Actor adopts normal Plan for Goal	Get **Actor** to adopt other plan
Spontaneous	
Actor in Situation	Change or avoid situation
— elicits — **Actor** does Act	Break "Situation-->Act" response
Emotional	
Situation evokes Emotion for **Actor**	Modify emotion felt by **Actor**
— elicits — **Actor** does Act	Break "Emotion-->Act" response
Conventional	
Actor in Situation repeatedly	Keep agent from Situation
— elicits — **Actor** does Act	Break "Situation-->Act" response
Actor believes Act is socially common or acceptable	Change **Actor's** belief in social convention
Habitual	
Actor in Situation repeatedly	Keep agent from Situation
— elicits — **Actor** does Act	Break "Situation-->Act" response

Figure 3: Models and Strategies for **Internal** Volitions

lematic action as external, and identified some agent as **motivator**, can you apply modification strategies that seek to influence the **motivator** so he changes his demands on the **actor**. For internal volitions you must focus on the **actor** alone, (although you can, of course, in your counter-planning, act as **motivator** and perhaps introduce a **third-party**). Only when volition assessment identifies a **third-party** can you design interventions that modify the **third-party's** role and its effect on the **actor's** behavior.

Figure 3 shows each of the internal volitions. They should all be read as specializations of the single table for internal volitions in Figure 2. These volitions illustrate differences on both the dimensions of "influence" and "choice." Lack of choice can result from lack of knowledge; in that case, we can augment the **actor's** knowledge. If the only way John knows to discipline his son is to beat him

MODEL	STRATEGY
Uncontrolled	
Actor in Situation	Avoid situation
— results-in — **Actor** does Act	Change situation (disenable causation)
Unknowing	
Actor missing information about Act	Inform **Actor** about Act
Uninvolved	
Actor did not actually do Act	Find out who (if anyone) did
Compelled	
Motivator do some Act	Prevent **Motivator's** Act
— results-in — **Actor** do Act	Change **Motivator's** Act (disenable causation)
Misled	
Motivator do some Act	Prevent **Motivator's** Act
— results-in — **Actor** misled	Change **Motivator's** Act (disable causation)
Actor missing information about Act	Inform **Actor** about Act
Framed	
Motivator do some Act	Prevent **Motivator's** Act
— results-in — **Advisor** misled	Change **Motivator's** Act (disable causation)
Actor did not actually do Act	Find out who did (suspect **Motivator**)

Figure 4: Models and Strategies for **Non-Volitional** Volitions

silly, we might suggest he try restricting privileges instead. Similarly, if Mary knows about mass transit, but chooses to drive to work because she doesn't know about the bus that runs near her house, we can open up new options by giving new specific information instead of giving a completely new plan.

Figure 4 shows the causal models and modification strategies underlying all the unconventional, "non-volitional" volitions. These reflect a total lack of choice on the **actor's** part. Obviously, if there was no choice about the action then it makes no sense to try and change it by appeal to the **actor's** intentional mechanisms: we can't offer an alternate goal, we can't threaten or cajole. Instead we either have to change the circumstances or we have to work on the **motivator** if we can find one. You can't convince someone not to sneeze even if you want them to keep quiet; better to remove the dust. If a baby cries you probably have to look to the parent for a way to get it to stop.

In turning to the external volitions shown in Figure 5, the dimension of "influences" becomes most salient. For external volitions the *strength* of the influences generally increases as you progress through the list of persuasion plans. Simply asking is clearly the weakest. Irrational appeal implicit in **inspiration**, if accepted, will override the rational argumentation of **convince**. Self interest is generally the strongest persuasion. There is a tendency to engage more important goals, (and themes), or invoke stronger impacts as you progress from **obliged** to **bought** to **threatened**. To influence any of the agents involved in causing an action you can try to establish any of the forms of external volition. Which ones will actually work, (and with what specific inducements), will depend on the details of the situation, but in general, escalating to a higher level of persuasion is a good strategy [9].

Beyond the general strategy of trying a stronger form of persuasion, many volition classes suggest relatively specific strategies. As the John and Mary examples illustrated, if you classify an **actor's** volition as **default**, you may be able to change the behavior simply by offering another option. If you believe the behavior resulted spontaneously from an **emotional** state, you may be able to set up a countervailing emotion. **Inspiration** may be counteracted by undermining the role-model. Actions performed in response to **threats** may be changed by offering protection. Actions resulting from **compulsion** can be changed by disabling the application of force. **Actors** who **misunderstand** what they are doing can be enlightened; those who are **misled** can be warned of the trickery.

Finally, Figure 6 shows the models and strategies for the third-party volitions. These are simply more specific versions of the external volitions. For example both **hostage** and **blackmail** situations can be viewed as a **threats**. But of course these situations offer more options because of the additional agent involved.

Model	Strategy
Persuade	
Motivator Persuades **Actor**:	Block transmission or receipt of Persuasion
"**Actor** do Act for Inducement"	Change contents of transmitted Persuasion
Actor believes Inducement valid	Convince **Actor** Inducement is false Convince **Actor** Inducement will be blocked
Actor values Inducement	Convince **Actor** Inducement is unimportant
Actor values Inducement more than consequences of Act	Convince **Actor** Inducement is less important
Actor agrees to do Act due to Inducement	Use stronger Persuade than **Motivator's** to get **Actor** to refuse
— motivates — **Actor** does Act	Use stronger Persuade than **Motivator's** to get **Actor** to renege on agreement
Asked	
Convinced	
Actor believes arguments	Undermine **Actor's** belief in validity or truth of arguments
Inspired	
Actor respects **Motivator**	Undermine **Actor's** respect for **Motivator**
Obliged	
Actor believes he has unfulfilled obligation to **Motivator**	Convince **Actor** there is no obligation to **Motivator** Convince **Actor** obligation to **Motivator** is already fulfilled
Bought	
Threatened	

Figure 5: Models and Strategies for **Persuade** Volitions

Model	Strategy
Surrogate	
Motivator influences **Third-Party**	Change **Motivator's** influence on **Third-Party**
Third-Party influences **Actor**	Change **Third-Party's** influence on **Actor**
Administer	
Motivator Incites **Actor**	Block transmission or receipt of Incitement
Third-Party delivers Inducement	Block execution of Inducement
Receive	
Actor believes **Third-Party** values Inducement	Convince **Actor** that **Third-Party** indifferent to Inducement
Actor believes **Third-Party** values Inducement more than consequences of Act	Convince **Actor** that **Third-Party** thinks Inducement less important
Actor agrees to do Act due to Inducement	Offer extra inducement for **Third-Party** to get **Actor** to refuse
— motivates — **Actor** does Act	Offer extra inducement for **Third-Party** to get **Actor** to renege on agreement

Figure 6: Models and Strategies for **Third-Party** Volitions

6 Using the Strategies

We can now return to the problem that opened this paper and suggest advice for the wife seeking to reclaim her possibly errant husband. Each item of the following list offers instantiations of strategies for the corresponding situation described earlier. The bracketed boldface word is the volitional classification:

1. **[Premeditated]** The wife can threaten her husband with divorce. Alternately, she can try to buy or bully the mistress into surrendering her claim. Of course, she can use any of the third-party techniques, perhaps attempting to bring the mistress to heel through the offices of *her* husband, or family, or priest.
2. **[Emotional]** The wife can lobby that her husband not go on so many long business trips alone, or arrange that they stay in closer touch when he is on the road.
3. **[Asked]** The wife can simply ask her husband not to spend too much time hanging around with other women, even as a harmless favor to a friend.
4. **[Obliged]** The wife can try to override her husband's obligation, but she has a stake in not ruining his standing at work. She can try to influence the boss somehow, but again, she can't afford to threaten and probably doesn't have much to offer.
5. **[Uninvolved]** There is nothing to do in this case, since nothing really happened.

Notice that detailed advice depends on specifics of the plans and actions chosen to address active goals, and on the specific mechanisms that introduce problems. Experience with specific problems (and past solutions) may often provide surer results than general reasoning through abstract strategies. When available it is preferable to volitional analysis. Of course, if such specific advice is found to need modification in order to fit the current circumstances, we are back in the situation of a case-based planner looking for plan repair rules, and volition may have a role to play.

The classification of volitions proposed in this paper is intended to serve as a useful abstraction summarizing the causation of actions in a way that discriminates among strategies for modifying those actions. Application of the strategies may not be possible without understanding the full motivation underlying these situations. The ABBY system has a model of the social domain that includes knowledge of social units, interpersonal themes, resulting goals, relevant plans, specific social actions, and the effects of those actions on identified goals. Volitional analysis is just one influence on its selection of advice, but this analysis, and the strategies it picks out, hold an important place in the arsenal of case-based reasoning techniques for the social domain.

Acknowledgements

Thanks to Eric Jones for many useful discussions and comments on several drafts of this paper. Thanks also to Kemi Jona for his comments on this paper.

References

[1] J.G. Carbonell. *Subjective Understanding: Computer Models of Belief Systems.* PhD thesis, Yale University, 1979.

[2] E. Charniak and D McDermott. *Artificial Intelligence.* Addison Wesley, Reading, MA, 1985.

[3] C.C. Collins. *Plan Creation: Using Strategies as Blueprints.* PhD thesis, Yale University, 1987.

[4] E.A. Domeshek. Understanding stories in their social context. In *Proceedings of the Tenth Annual Conference of the Cognitive Science Society,* Montreal, August 1988. COGSCI-88.

[5] G.D. Dyer. *In-depth Understanding: A Computer Model of Integrated Processing For Narrative Comprehension.* PhD thesis, Yale University, 1982.

[6] K.J. Hammond. *Case-based Planning: An Integrated Theory of Planning, Learning and Memory.* PhD thesis, Yale University, 1986.

[7] J. Kolodner. Extending problem solver capabilities through case-based inference. In *Proceedings of the Fourth International Workshop on Machine Learning,* pages 167–178, Los Altos, CA, June 1987. University of California, Irvine, Morgan Kaufman Publishers, Inc.

[8] J.L. Kolodner, R.L. Simpson, and K. Sycra-Cyranski. A process model of case-based reasoning in problem-solving. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence,* Los Angeles, CA., August 1985. IJCAI.

[9] R.C. Schank and R. Abelson. *Scripts, Plans, Goals, and Understanding.* Earlbaum, Hillsdale, N.J., 1977.

[10] C.M. Seifert. *Mental Representations of Social Knowledge: A Computational Approach to Reasoning About Relationships.* PhD thesis, Yale University, 1987.

[11] C.M. Seifert. Planning principles specific to mutual goals. In *Proceedings of the Ninth Annual Conference of the Cognitive Science Society,* Seattle, August 1987. COGSCI-87.

[12] R.G. Simmons. *Combining Associational and Causal Reasoning to Solve Interpretation and Planning Problems.* PhD thesis, MIT, 1988.

[13] R. Wilensky. *Planing and Understanding.* Addison-Wesley, Reading, MA, 1983.

A Connectionist Approach to High-Level Cognitive Modeling

Rainer Goebel
Department of Psychology, Gutenbergstr. 18
3550 Marburg, West-Germany

Abstract—In this paper a connectionist framework is outlined which combines the advantages of symbolic and parallel distributed processing. With regard to the acquisition of cognitive skills of adult humans, symbolic computation is stronger related to the early stages of performance whereas parallel distributed processing is related to later, highly practiced, performance. In order to model skill acquisition, two interacting connectionist systems are developed. The first system is able to implement symbolic data structures: it reliably stores and retrieves distributed activity patterns. It also can be used to match in parallel one activity pattern to all other stored patterns. This leads to an efficient solution of the variable binding problem and to parallel rule matching. A disadvantage of this system is that it can only focus on a fixed amount of knowledge at each moment in time.
The second system – consisting of recurrent back-propagation networks – can be trained to process and to produce sequences of elements. After sufficient training with examples it possesses all advantages of parallel distributed processing, e. g., the direct application of knowledge without interpreting mechanisms. In contrast to the first system, these networks can learn to hold sequentially presented information of varying length simultaneously active in a highly distributed (superimposed) manner. In earlier systems – like the model of past-tense learning by Rumelhart and McClelland – such forms of encodings had to be done "by hand" with much human effort. These networks are also compared with the tensor product representation used by Smolensky.

1 Introduction

Parallel distributed processing seems especially well suited to model automatic, subconscious information processing. Symbolic computations for example variable binding, explicit rule following and deliberate planning are much harder to achieve.
I think that for modeling high-level cognitive processing both styles of computation are necessary. Considering the *temporal dimension* of learning a cognitive skill reveals that they are even intimately related: early in acquisition, symbolic processing is very important because instructions (facts and rules) must be stored and processed deliberately. The application of knowledge during this stage is controlled, slow, effortful and generally serial but it has the great advantage that approximate performance is achieved immediately through general *interpreting* procedures.
In later stages of practice the relevance of connectionist computation gradually increases: the explicit, declarative knowledge is converted to more flexible, more robust and richer procedural knowledge. The application of this knowledge is for the most part *automatic*, requires few attention, is fast and more parallel: it constitutes "expert knowledge". This kind of *directly* applied knowledge gradually emerges because subskills are automated: elementary processing steps are associated together and activated as whole units.

According to this view, I propose a connectionist framework which considers the acquisition of cognitive skills in terms of two interacting components (similar to Norman, 1986): one component – capable of symbol processing – trains or 'programs' the other component. The first component consists of 'symbol-modules' which efficiently implement essential aspects of conventional (von Neumann) computers. The second component consists of 'PDP-modules' – recurrent back-propagation networks – which possess all advantages of parallel distributed processing.

I will explain the basic features of the proposed system using a simple illustration task.

2 The Illustration Task

The proposed framework addresses skill acquisition of adults or at least older children who sufficiently master their native language. To explain the basic ideas of the approach, I use a very simple illustration task: 'Pig Latin' (Harvey, 1985). Children often learn to speak Pig Latin as a "secret" language. A word is translated into Pig Latin according to the following rule: take any consonants at the beginning (up to the first vowel) and move them to the end. Then add "ay" to the end. So CAT becomes ATCAY, TRUST becomes USTTRAY and IS becomes ISAY. To make things a little bit more difficult, I added the following rule: if the word begins and ends with a vowel then add "way" to the end. Thus EASY doesn't become EASYAY but EASYWAY which also sounds better. Note that Pig Latin is spoken, thus pronounciation is relevant for applying the rules. In the following however the rules are applied to strings of letters. Despite the simplicity of these rules, it requires some practice until one speak Pig Latin fluently!

After general remarks about the used networks the basic features of symbol-modules are explained. Then it is outlined how a symbol module can explicitly follow the Pig Latin rules. Thereafter PDP-modules are described and it is shown how a special module learns by example to behave as following the rules. Finally it is outlined how both modules are integrated in such a way that the symbol module trains the PDP-module to follow the rules.

3 Recurrent Networks: Handling Temporal Data

Cognitive processes evolve in time: sequences of elements (for example letters, phonemes or words) must be processed and produced as output. Therefore both modules consist of (multilayered) *recurrent* networks. Recurrent networks are well suited to handle temporal data because they provide a kind of short term memory for retaining activity states over time. The system operates in discrete time steps; the update scheme is as follows: layers are updated synchronously in ascending order (as in feedforward nets). If unit i is updated at time step t and there are recurrent connections (from the same or higher layers) to unit i, then the activity of the corresponding units at time step $t-1$ is used. Both modules use linear and semi-linear units.
PDP-modules are trained using the generalized delta rule. Back-propagation allows networks to find themselves interesting internal representations, but is designed for feedforward nets only. Rumelhart et. al. (1986) describe a technique which allows the application of back-propagation to arbitrary connected networks: Every recurrent network operating in (discrete) time events can be "*unfolded in time*" to a feedforward network with identical behavior. However this technique has some disadvantages. Therefore I made some modifications yielding a system with high efficiency (see Goebel, 1990)

4 Symbol-Modules: Explicit Rule Following

Symbol-modules are special architectures designed to store and retrieve distributed activation patterns (symbols) reliably and immediately.
A basic assumption, also valid for PDP-modules, is that the activity of some units can serve as target values for nearby output units. This allows one module to send teaching signals to other modules.
In its simplest form a symbol module is composed of two parts (see figure 1). One part — consisting of two equal width layers — acts as a 'gate' for symbols. Through this 'symbol gate' the module can receive symbols from other modules or from input units. It also can send symbols through this gate to other modules or to output units. At each time step one module can handle one symbol.
The other part consists of a (large) layer that acts as a sequence of 'memory cells'. These 'local memory units' are totally connected to the output units.

4.1 Location-Addressed Retrieval

Assume that a sequence of symbols, for example the string TIME is to be stored in the symbol-module of figure 1. Each (linear) local memory unit has a connection to its right neighbor with weight 1. In this example the number of units N_s that constitute a symbol is 5.

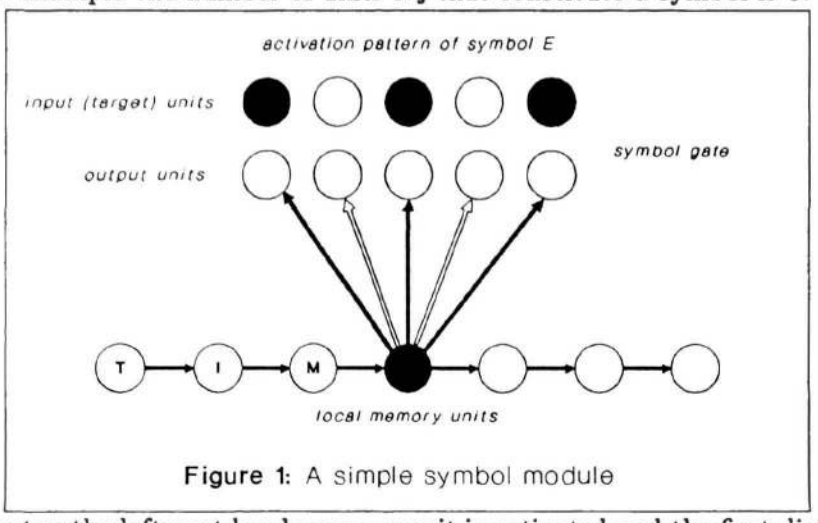


Figure 1: A simple symbol module

At the first time step the leftmost local memory unit is activated and the first distributed pattern – representing T – appears at the target units of the symbol gate. Now learning takes place according to the delta rule. The result is that the activation values of the target units are mapped to the weights coming from the active local memory unit to the output units.
At the next time step the second local memory unit is active and the second symbol (I) appears on the target units. Now these activation values are mapped to the corresponding weights etc.
Consider retrieving the sequence. Assume all local memory units are turned off. Now the leftmost local unit is turned on and activation flows through the learned weights to the output units evoking the first stored symbol T. At the next time step the next local unit is active evoking the second stored symbol I and so on. This is *location-addressed* retrieval.
The delta rule changes the weights as follows:

$$w_{ol}(t+1) = w_{ol}(t) + \epsilon a_l(a_t - a_o)$$

The value of learning rate ϵ is 1. (The index l stands for a local unit, o for an output unit and t for a target (input) unit.) Consider the weights between a non-active local unit and the output units:

$$w_{ol}(t+1) = w_{ol}(t) + 0(a_t - a_o) = w_{ol}(t)$$

This says that the weights between non-active local units and the output units are not affected by the learning process. For the weights between the active local unit and the output units results:

$$w_{ol}(t+1) = w_{ol}(t) + 1(a_t - a_o) = w_{ol}(t) + a_t - a_o$$

Note that a_o is the value of a linear output unit: $a_o = a_l w_{ol}(t)$. Because in the considered case $a_l = 1$ we have

$$w_{ol}(t+1) = w_{ol}(t) + a_t - w_{ol}(t) = a_t$$

This means: regardless of the current weight between the active local unit and an output unit, the new weight equals exactly the activation value of the corresponding target unit.

According to this learning scheme each distributed symbol is stored at a distinct location (has its own weights) and is therefore not affected by other symbols in memory. The result is a dualistic representation scheme: each symbol is represented both in a local and a distributed form.

I have improved this simple symbol module in many ways, for example introducing 'pointers' to control the activation of local memory units. This allows to build symbolic data structures like stacks and lists. (The first larger system I have build wiht symbol modules is a simple LISP interpreter.)

4.2 Content-Addressed Retrieval

Suppose that the layer of local memory units and the layer of output units are totally connected in *both* directions. Symbols consist of $1, -1$ patterns. The storing mechanism is as before but each dictated weight change now operates in both directions yielding symmetrical connections: $w_{ol} = w_{lo}$.
Such symbol modules can be used from two sides to retrieve stored symbols. In addition to location-addressed retrieval (from the local units to the output units), *content-addressed* retrieval from the output units (acting as input units) to the local memory units is possible.
Content-addressed retrieval works as follows: If symbol s_i, for example M, is presented to the output units, activation flows down to the local units. Because the weights to each local unit represent a stored symbol, the current activation pattern s_i is compared in parallel to all stored symbols! The net input to local unit l_j reflects the strength of match between s_i and s_j.
If each local unit has a threshold of $N_s - 2$ the symbol(s) which *exactly matches* s_i become active. If the local units are designed as a Winner-Take-All (WTA) network, the *best matching* symbol is retrieved. Best match also allows *pattern completion* if some output units are clamped to 0.

The combination of content- and location-addressed retrieval can be used to bind values to variables. Suppose the sequence 'X,3,Y,5,Z,7' is stored in a symbol module. Now to retrieve the value of variable 'Y', one just has to present its pattern at the output units. This activates its corresponding local memory unit. At the next time step its right neighbor is active producing the corresponding value. Thus retrieving the value of a variable operates in constant time, independent of the number of stored variables. I have enhanced this variable binding mechanism to bind sequences of elements and to allow to retrieve the value of the value of a variable.

4.2.1 Parallel Rule Matching

On the basis of the described features of symbol modules, I have developed a simple production system which processes the Pig Latin rules (see figure 2).

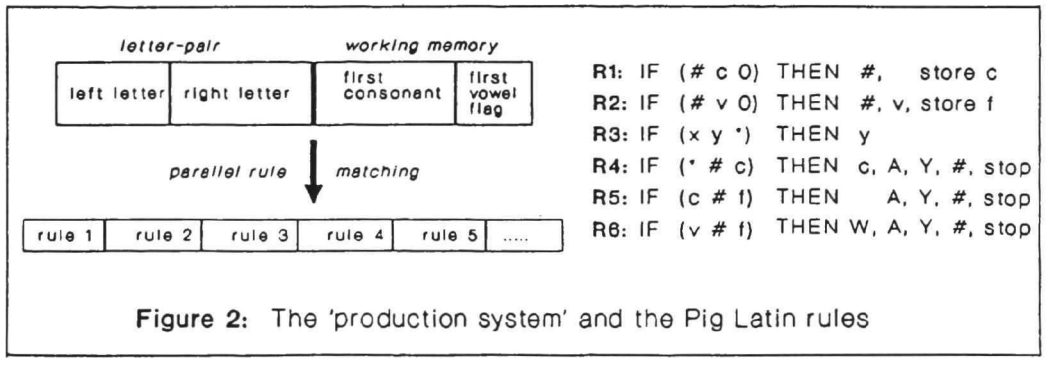


Figure 2: The 'production system' and the Pig Latin rules

This special symbol module has a larger symbol gate to represent two letters and some further information (working memory). A condition is represented through one local unit. In this version exact match is used. No (local) variables are yet established. If a local unit becomes active, it activates its right neigbors constituting the action part. Actions can send symbols to a special output module or store values in working memory. There is no conflict resolution yet implemented, thus the rules must be defined such that the conditions exclude each other.
There are several ways to define the Pig Latin rules within the scope of this production system.

Since no local variables are yet allowed, I have restricted the rules in the following way: only three consonants (B, M, T) and two vowels (E, I) are used. Additionally, if a word begins with a consonant, the next letter must be a vowel. For reasons which will become clear in the next section a string is not presented as a sequence of letters, but in small chunks, consisting of letter-pairs. Word-edges are indicated with the special symbol '#'. Thus the string #TIME# is actually presented as the following sequence: #T, TI, IM, ME, E#.
The rules for the production system are shown in figure 2. The 'c' stands for a consonant, 'v' for a vowel and 'x' and 'y' for a vowel or a consonant but not #. The letter f represents a flag in working memory indicating that the first letter was a vowel. The '*' indicates that the short term memory may have an arbitrary value. Note that each rule is actually repeated several times with local variables replaced with letters. Thus an actual rule as an instance of the first rule is: 'IF (# B 0) THEN #, store B'.
Consider as an example how the string #TIME# is processed. At first, #T is presented to the output units. Now all conditons are compared in parallel against the data. The first rule matches, sending # to the output module and stores T in the short term memory. Now the next pair TI is presented and rule 3 matches, sending I to the output module. The next pairs IM and ME are treated the same. Then the last pair, E#, is presented. Rule 4 matches, finally sending T, A, Y and # to the output module.

5 PDP-Modules: Learning by Example

PDP-modules are ordinary (recurrent) back-propagation networks, thus possessing their main advantages: simultaneous consideration of many pieces of information, learning from experience and generalization to novel situations. To exploit these features of parallel distributed processing, a different approach to learn the Pig Latin rules is reasonable: present a whole input string at once to an input layer and compute in one step, in parallel, the correct output string.

But there is a problem: how are strings – entities of varying length – represented with a fixed number of units? One solution might be to use a buffer large enough to hold the longest string: the input layer and output layer are divided in many parts ('slots'), each representing one element (letter) in successive order. But there are some problems with this *position dependent* representation (Mozer, 1988). The most important problem with regard to the Pig Latin task is that this representation does not support proper generalization because it cannot handle word edges – the most critical information – in isolation, independant from the length of the string.
Rumelhart and McClelland (1986) solved this problem within the scope of the past tense model using a *context-sensitive* representation: each element (phoneme) is represented together with its predecessor and its successor constituting a "Wickelphone". To represent the word 'represent', it is decomposed into the following set of triplets: #Re, rEp, ePr, pRe, rEs, eSe, sEn, eNt, nT#. Now each Wickelphone represents a 'slot' (one input unit) and all words ending in the same phonemes have the same slot active! This allows to extract the relevant regularities.

According to this representation scheme I have implemented a simple two layer network to process strings according to the Pig Latin rules. To keep things simple only 'Wickel-pairs' are used. Of course, letter-pairs aren't well suited to represent strings distinctly (Also Wickelphones cannot represent every string distinctly; see Pinker & Prince (1988) for a thorough criticism of the past tense model). Therefore I have defined the additional constraint that no letter may appear twice. This constraint together with those of the previous section lead to 39 letter-pairs (39 input and 39 output units) and 71 distinct strings.
The network learns the Pig Latin task quickly. After training with 50% of the strings it generalizes well to almost all remaining strings (only three strings are processed incorrect).
This solution however has two serious drawbacks. First, if more letters are allowed to build strings, there are quickly too many letter-pairs (this was the main reason to use 'Wickelfeatures' – a more compact, distributed representation – in the past tense model). Second, a simple feedforward network cannot handle sequentially presented data.

What one would like is a network which combines elegantly the strengths of both parallel and serial processing. Therefore I have developed a PDP-module which can sequentially process letter-pairs and can hold the presented information simultaneously active in a highly distributed (superimposed) manner.

In the following, the basic idea is described using local representations of the pairs in a pre-wired simple network. Then it is outlined how the network can learn itself proper representations.

Figure 3 shows a small illustration network. The input layer and the output layer consist of two 'slots' each representing one of three elements (A, B or C). The hidden layer consists of six local units representing letter-pairs. Now suppose the elements B and C are presented to the input layer. They activate the pair unit BC which in turn activates the elements B and C of the output layer (see figure 3). Each (linear) pair unit is self connected with a value of +0.8. Thus, the activity of the active pair unit slowly decays as time proceeds and consequently the activity of the output units, too. But this does not really happen because of recurrent connections from the left three output units to the pair units. These output units have positive connections to pairs which 'match' the output unit with its left side. The output unit representing B, for example, has positive connections to all pairs Bx and zero weights to all other pairs.

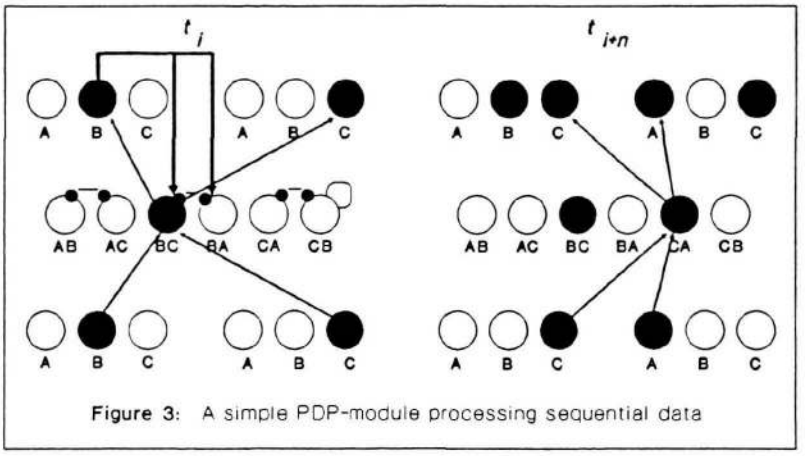


Figure 3: A simple PDP-module processing sequential data

Thus at the next time step the active output unit B activates the pair units BA and BC. This prevents the pair unit BC from decaying to zero but BA gets some activity, too. However each pair unit has inhibitory connections to other pair units with the same left element. This allows BC to prevent BA from becoming active. In summary, the activity state of the pair units at time step $t+1$ is the same as before. The active pair unit and the active output unit mutually reinforce each other as time proceeds creating a stable 'resonant state' (Grossberg, 1987).

Why not using self connections of value +1? In this case no recurrent connections from the left output units would be necessary. The basic trick however is that the same left output unit can activate different units of the right output units depending on the stored pair units. Thus the system acts as a 'dynamic pattern associator': the mapping function from left to right is modulated by the activity of the pair units!

Now suppose the next letters C, A are presented; they turn on the CA pair unit which itself activates the corresponding output units (see figure 3). Through resonance both pair units remain active. Note, that from looking only to the output units one cannot decide whether B goes with C and C with A or whether B goes with A and C with C! This is essentially the variable binding problem (Smolensky, 1987). The right information is present in the hidden layer of pair units. Retrieving that knowledge operates as follows: all left output units are clamped to 0 except the unit in question. If for example A and C are clamped to zero, only the pair BC gets support from B, CA starts to decay. Consequently the activity of the A-unit on the right side also decreases. Thus only the left B and the right C remains active!

This small network was entirely "hand-wired" to explain the basic idea. Is it possible to make a network learn the right weights on its own? The answer is yes, if the network structure is restricted in some ways. PDP networks do generalize to novel inputs but there are often too many generalization functions. A possible solution to this problem is to build some a priori ('innate')

knowledge into the network, guided from the solution space the network might discover.
I have replaced the fixed weights in the described network with random weights but imposed the following constraints: only positive connections are allowed from lower to higher layers and also from the left output units to the hidden layer. Within the hidden layer only inhibitory connections are allowed except the self-connections which are fixed at 0.8.
The network is trained with all pairs. At the first time step, two letters are presented to the network and also used as target values for the output units. At the next time step, the hidden units gets its own previous activations as target values. According to this training regime, the network tries to find distributed representations 'rLR' in the hidden layer which must suffice three conditions: L + R → rLR (input to hidden); rLR → L + R + r'LR (hidden to output and hidden to hidden); L + r'AB → rAB (left output and (decayed) hidden to the same hidden as before). Note that only single pairs are used to train the network but the imposed constraints strongly suggest generalization to hold more than one pair active.

This network finds interesting distributed representations for the letter pairs. If strings are presented sequentially the distributed representations of the letter-pairs are superimposed upon each other. The network has no fixed capacity to hold letter pairs but saturates gracefully with the number of stored pairs. The capacity can be determined through the number of used hidden units (the capacity depends upon the alphabet size and the number of units, see Rosenfeld & Touretzky, 1988). This is in contrast to the variable binding mechanism – the tensor product representation – used by Smolensky (1987) which has a fixed capacity.

In summary, the final PDP-module processes sequences of letter pairs constructing a highly distributed representation which produces a corresponding representation of the output string. This representation in turn can be decomposed to the elemantary letters. Thus it can handle sequential input and output like the symbol module.

6 The Synthesis: Rules and Practice

In the last sections it was shown how the Pig Latin rules can be taught to symbol-modules and PDP-modules. Both modules solve the problem exploiting their special advantages. To combine the virtues of each module, they are finally integrated into a larger system which operates as follows (see figure 4):

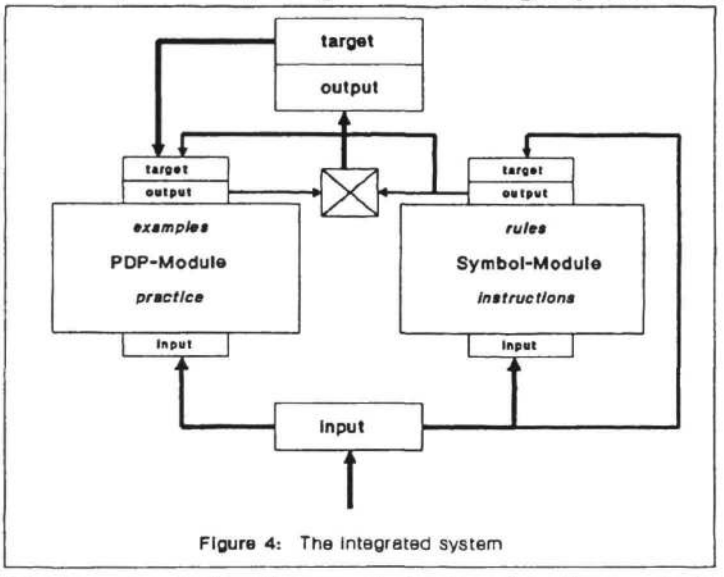


Figure 4: The integrated system

First, the Pig Latin rules are presented and successively stored in the symbol module. The system is now able to process strings according to the rules. The resulting strings obtained by the symbol module are send to the target units of the PDP-module. Therefore the PDP-module gradually learns to mimic the behavior of the symbol module and extracting the underlying regularities.
Consider a more interesting case: The Pig Latin rules are presented to the symbol module but not the special rule which handles strings both beginning and ending with a vowel. This case resembles

more closely to real life situations where instructions cover only the most strong regularities but not all subtle details. Now the symbol module operates correctly most of the time. Only if a string beginning and ending with a vowel is to be processed the symbol module produces an incorrect output string. In this case it is assumed that the environment provides an external teaching input which is transferred to the PDP-module. Thus the PDP-module is trained from two sources: most of the time from the symbol module (if it produces correct output) and sometimes from an external teacher yealding finally to more accurate and fluent performance than the symbol module.

7 Conclusions

This paper presents a suggestive approach to high-level cognitive modeling: a connectionist system learning both like traditional AI-systems, emphasizing rule-like knowledge, and also like PDP models, emphasizing learning through experience.
The concrete realization of this idea poses many new questions about the interactions of the proposed modules. The first results obtained are preliminary in nature but they promise that more complex systems will lead to attractive psychological models of skill acquisition and also offer new ways to escape the 'brittleness' of many conventional AI-systems.

Acknowledgements

Many thanks to Hans-Henning Schulze for helpful conversations during the writing of this paper. Thanks too to Thomas Göttsche, Klaus Hahn, Dieter Heim, Dirk Vorberg and Kai-Uwe Wagner.

References

Goebel, R. (1990). Learning Symbol Processing with Recurrent Networks. In R. Eckmiller (Ed.), Proceedings of the International Conference on Parallel Processing in Neural Systems and Computers (ICNC), Düsseldorf, F.R.G., 19-21 March, 1990.

Grossberg, S. (1987). Competitive Learning: From Interactive Activation to Adaptive Resonance. Cognitive Science, **11**, 23-63.

Harvey, B. (1985). Computer Science Logo Style. Volume I, MIT Press, Cambridge, MA.

Mozer, M.C. (1988). A Focused Back-Propagation Algorithm for Temporal Pattern Recognition (Tech. Rep.). University of Toronto, Departments of Psychology and Computer Science

Norman, D. A. (1986). Reflections on Cognition and Parallel Distributed Processing. In D. E. Rumelhart and J. L. McClelland (Eds.), Parallel Distributed Processing. Volume II, MIT Press, Cambridge, MA.

Pinker, S. & Prince, A. (1988). On Language and Connectionism: Analysis of a Parallel Distributed Processing Model of Language Acquisition. Cognition, **28**, 73-193.

Rosenfeld, R. & Touretzky, D. S. (1988). A Survey of Coarse-Coded Symbol Memories. In D. S. Touretzky, G. Hinton and T. Sejnowski (Eds), Proceedings of the 1988 Connectionist Models Summer School.

Rumelhart, D.E., Hinton, G.E. & Williams, R.J. (1986). Learning Internal Representations by Error Propagation. In D. E. Rumelhart and J. L. McClelland (Eds.), Parallel Distributed Processing. Volume I, MIT Press, Cambridge, MA.

Rumelhart, D.E., & McClelland, J.L. (1986). On Learning the Past Tenses of English Verbs. In D. E. Rumelhart and J. L. McClelland (Eds.), Parallel Distributed Processing. Volume II, MIT Press, Cambridge, MA.

Smolensky, P. (1987). On Variable Binding and the Representation of Symbolic Structures in Connectionist Systems (Tech. Rep.). University of Colorado at Boulder, Department of Computer Science.

Fuzzy Implication Formation in Distributed Associative Memory

Rick L. Jenison and Gregg C. Oden
University of Wisconsin

An analysis is presented of the emergence of implicational relations within associative memory systems. Implication is first formulated within the framework of Zadeh's theory of approximate reasoning. In this framework, implication is seen to be a fuzzy relation holding between linguistic variables, that is, variables taking linguistic terms (e.g., "young", "very old") as values. The conditional expressions that obtain from this formulation may be naturally cast in terms of vectors and matrices representing the membership functions of the fuzzy sets that, in turn, represent the various linguistic terms and fuzzy relations. The resulting linear algebraic equations are shown to directly correspond to those that specify the operation of certain distributed associative connectionist memory systems. In terms of this correspondence, implication as a fuzzy relation can be seen to arise within the associative memory by means of the operation of standard unsupervised learning procedures. That is, implication emerges as a simple and direct result of experience with instances of events over which the implicational relationship applies. This is illustrated with an example of emergent implication in a natural coarsely coded sensory system. The percepts implied by sensory inputs in this example are seen to exhibit properties that have, in fact, been observed in the system in nature. Thus, the approach appears to have promise for accounting for the induction of implicational structures in cognitive systems.

The nature of implication is a contentious topic in the philosophy and psychology of reasoning. Much of the difficulty arises from attempts to make implication be truth functional and, thus, can be avoided by recognizing that, in most instances, conditional propositions are atomic. Nevertheless, like any other bit of knowledge, such propositions must arise from experience somehow. The present paper discusses how the induction of implication may be accomplished through self-organizing processes in a distributed associative memory.

Linguistic variables, fuzzy relations and the compositional rule of inference

It will be useful to consider the very general form of implication developed by Zadeh (1975) as part of a system for approximate reasoning. This system is based on the concept of linguistic variables. A linguistic variable takes on values that are words or phrases rather than numbers. An often cited example is that of the linguistic variable *age* which may have as values "old", "very old", "not very old", "young", etc. Each such value is actually a fuzzy set defined over the underlying base variable of years lived. That is, each linguistic value is fuzzy in that it may be appropriately applied to various years to different degrees. For any given underlying value (number of years), there will be some degree to which it belongs to the fuzzy set. Figure 1 gives illustrative membership or compatibility functions for "young", for "old" and for "very old" (see Oden, 1978, 1984).

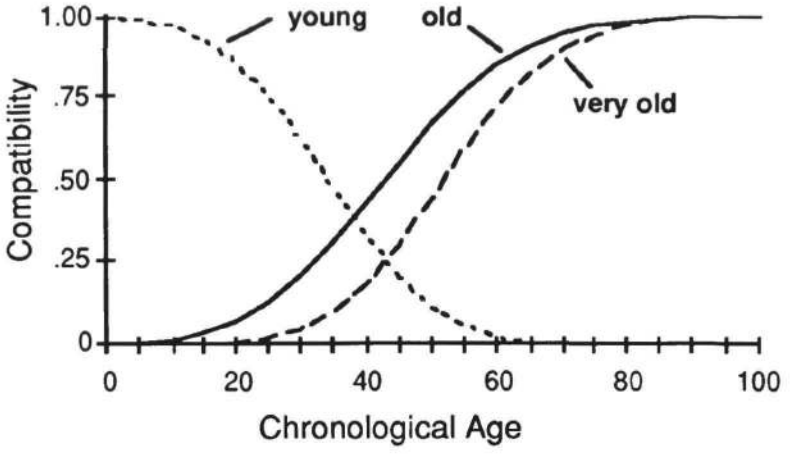


Fig. 1. Compatibility of chronological ages with various linguistic values.

Another way to view the notion of linguistic variables is by means of a hierarchical representation such as that shown in Figure 2.

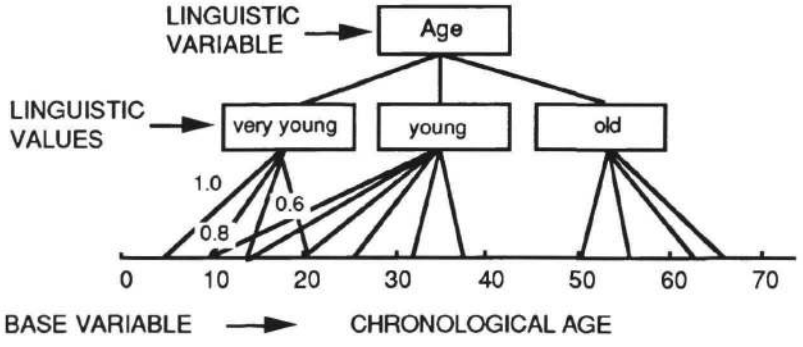


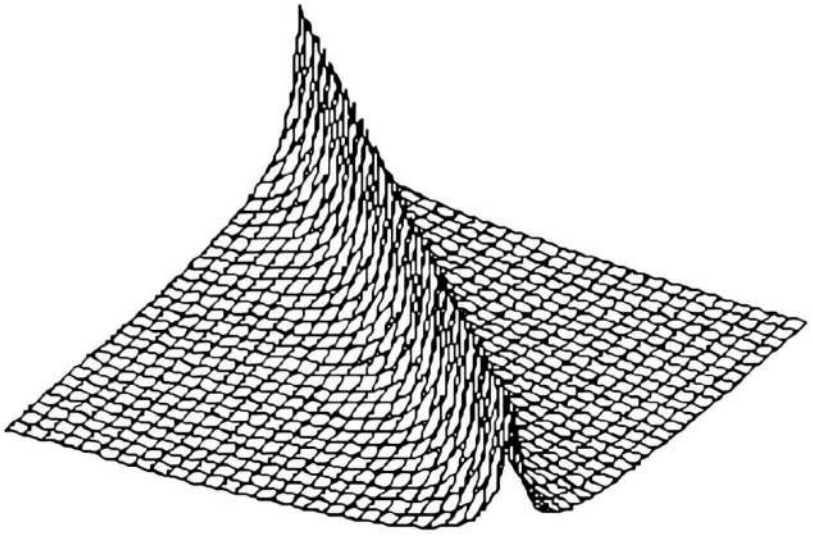
Fig. 2. Hierarchical structure of a linguistic variable.

This shows that the linguistic variable "age" can take on any of several linguistic terms as its value and that each term, such as "very young", is a fuzzy set containing each of the possible values of the base variable (chronological age) to various degrees. Again, the degree of membership of a given chronological age in the fuzzy set "very young" can be thought of as the compatibility of that age with the concept of a very young person. It is sometimes useful to represent fuzzy sets in terms of the pairings of each base variable value with its associated compatibility value

$$\text{very young} = \{(5, 1.0), (10, 0.8) \ldots (120, 0)\},$$

which is often more compactly represented in vector notation, A = [1.0 0.8 ... 0.0]. These representations require a discretized view of the base variable, but, of course, one can have as small a grain size as one wishes.

A second component of Zadeh's extension of implication involves the set-theoretic notion of relations holding between objects. For example, "less than" is a relation that holds between numbers. A *fuzzy* relation R holding between X and Y is a fuzzy subset of the Cartesian product $X \times Y$, which is often represented as a matrix having entries that are the respective membership degrees, denoted by $\mu_R(x,y)$. A useful way of visualizing a large complicated matrix, such as is required to represent a fuzzy relation, is to plot the matrix as a surface. Figure 3 illustrates a resemblance relation plotted as a surface and also in tabular form[1].



1.0000	0.5000	0.3333	0.2500	0.2000	0.1667	0.1429
0.5000	1.0000	0.5000	0.3333	0.2500	0.2000	0.1667
0.3333	0.5000	1.0000	0.5000	0.3333	0.2500	0.2000
0.2500	0.3333	0.5000	1.0000	0.5000	0.3333	0.2500
0.2000	0.2500	0.3333	0.5000	1.0000	0.5000	0.3333
0.1667	0.2000	0.2500	0.3333	0.5000	1.0000	0.5000
0.1429	0.1667	0.2000	0.2500	0.3333	0.5000	1.0000

Fig. 3. Relation $\mu_R(x,y)$ shown as a surface together with the upper left portion of its matrix.

Another way to represent a relation is as weighted graph or network, where $\mu_R(x,y)$ is taken to be the weight on the arc from each x to each y. When viewed in this fashion, one can imagine a complicated cascade of relations represented by a multilayered network architecture. The "glue" that joins relations is the composition operator denoted R o S. The fuzzy set-theoretic definition of composition is of a *union of intersections* over the membership functions $\mu_R(x,y)$ and $\mu_S(y,z)$. The operation is directly analogous to that of matrix multiplication in linear algebra.

---

[1] A resemblance relation is defined as being reflexive and symmetric, that is, one for which $\mu_R(x,y) = 1$ and $\mu_R(x,y) = \mu_R(y,x)$. The membership function shown in Figure 1 is $(1 - |x - y|)^{-1}$.

Zadeh draws the connection between fuzzy implication and fuzzy relations in terms of forming inferences based upon linguistic variables. Implication serves as a primitive for a basic rule of inference rather than as a connective derived from a truth table. Consider the conditional expression IF A AND $(A \Rightarrow B)$ THEN B, which defines modus ponens in traditional logic,. Here we infer the truth of proposition B from the truth of A and the truth of the implication $A \Rightarrow B$. Zadeh (1975) proposed an extension of modus ponens, which he called the compositional rule of inference, that uses the composition operator as described in the previous section. The compositional rule of inference is defined as follows: Let X and Y be two universes of discourse with base variables x and y, respectively. Further, let R(x), R(x,y) and R(y) denote fuzzy sets in X, $X \times Y$ and Y. Then the *compositional rule of inference* asserts that the solution of R(x) = A together with R(x,y) = F is given by R(y) = A o F. In this sense, R(y) is inferred from the combination of the linguistic value A and the relation F. For example, for

$$A = \text{small} = [\,1 \quad 0.6 \quad 0.2 \quad 0\,]$$

and

$$F = \text{approximately equal} = \begin{bmatrix} 1 & 0.5 & 0 & 0 \\ 0.5 & 1 & 0.5 & 0 \\ 0 & 0.5 & 1 & 0.5 \\ 0 & 0 & 0.5 & 1 \end{bmatrix}$$

then $R(y) = A \circ F = \text{small} \circ \text{approximately equal}$

$$= [\,1.0 \quad 0.6 \quad 0.2 \quad 0\,] \circ \begin{bmatrix} 1 & 0.5 & 0 & 0 \\ 0.5 & 1 & 0.5 & 0 \\ 0 & 0.5 & 1 & 0.5 \\ 0 & 0 & 0.5 & 1 \end{bmatrix}$$

$$= [\,1 \quad 0.6 \quad 0.5 \quad 0.2\,] = \text{more or less small.}$$

Distributed associative memory

In Zadeh's view, implication is a fuzzy relation and, thus, atomic rather than truth functional. The interpretation of implication as a primitive instead of as a connective, however, is still engineered rather than emergent. That is, the architect of the reasoning model describes the implication primitive in the form of a relation that reflects the designer's intuition. It is natural to wonder, however, how a particular implicational expression might arise naturally as a primitive term. One possible answer is based on the striking similarity of the compositional rule of inference to the distributed associative memory models proposed by Kohonen (1987). The linear recall problem as formulated by Kohonen is: what memory array allows one representation, b, to be recalled in response to another input representation, a, for a number of such associated a–b pairs? Kohonen formulates this issue in linear algebraic terms as: what is the matrix operator M by which a pattern b_k is obtained from a pattern a_k, i.e. $b_k = M \cdot a_k$? In the following discussion, the similarities of this question to compositional inference are explored.

For our purposes, associative memories are heteroassociative: they store associations of data from conjoint pairs: (a_1b_1), (a_2b_2),... (a_Pb_P), where $a = [a_1,...,a_N]^T$ and $b = [b_1,...,b_N]^T$ are vectors from different N-dimensional vector spaces . These vectors may correspond to crisp sets from binary vector space or fuzzy sets. By using a simple correlational or Hebbian rule, the associations between patterns a and b can be incrementally stored as an optimal orthogonal projection operator M, the memory matrix[2]. To illustrate how self-organization in associative memory systems may effect emergent implication structures, we will examine how coarse coding can be considered to be an implicational process. This discussion will be specifically developed in terms of sensory coding in the auditory system; this is simply because relevant neurophysiological data is available to us and we have developed a filter model of the auditory periphery (Jenison, Kluender, Greenberg, and Rhode, 1990). However, it should be emphasized that this is just a choice of convenience; this analysis can equally well be applied to coarse coding in any sensory or conceptual domain.

The peripheral auditory system can be thought of as a spectrum analyzer of sorts. That is, in the auditory periphery, frequency components are separated out and the degree to which each component contributes to the overall complex of sound is transmitted to higher processing centers by a pathway of nerve fibers. This mapping from a physical acoustic source to representation in the auditory system is not crisp by any means. A simple model of the acoustic to auditory mapping (derived from Peterson and Boll,1983) is shown schematically in Figure 5. This illustrates the coarse coding of the acoustic signal in terms of 23 auditory channels spanning the frequency range from 0 to 4500Hz (here quantized into 128 frequencies). Each channel is represented in the figure by a receptive field or response area mapped to the base variable of frequency. This is directly analogous to our discussion of linguistic variables for which membership functions specify the meaning of each linguistic value that may be assigned to a linguistic variable.

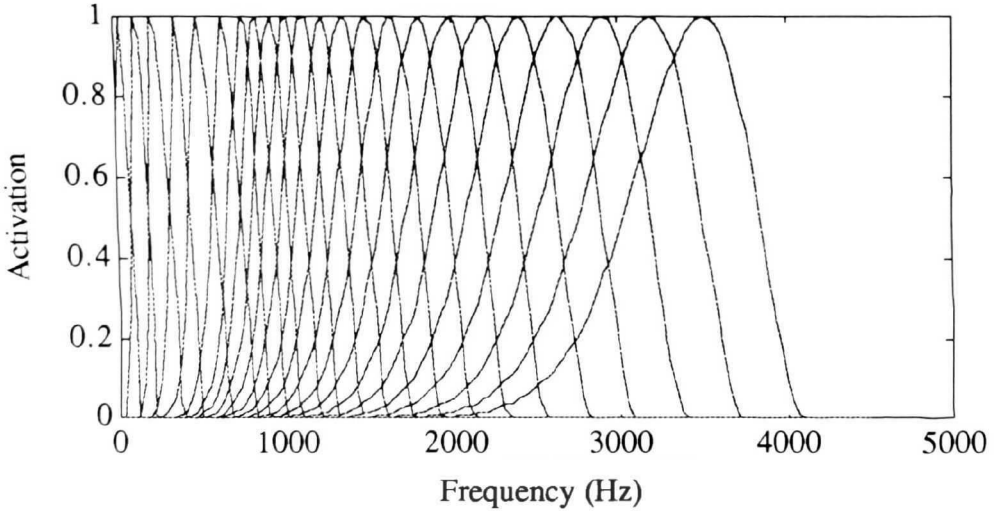


Fig. 5. Canonical auditory periphery response functions.

[2]Training the network M on the conjoint association of a and b turns out to be equivalent to solving M directly, $M = B \cdot A^+$, where $^+$ denotes the pseudoinverse operator (Kohonen, 1987). $A = [a^{(1)},..., a^{(P)}]$ and $B = [b^{(1)},..., b^{(P)}]$ denote the pattern matrices, where a and b are column vectors.

If we view the auditory response functions as *linguistic values* , how might implications form such that a perception is induced at higher cognitive levels? The auditory response functions shown in Figure 5 can be cast in matrix notation, where each response area corresponds to a row in the matrix. The column vectors, therefore, correspond to the excitation pattern across the 23 nerve fibers generated by a particular frequency component. Similarly, the output of the system is a matrix which corresponds to the identity matrix. The identity matrix is representative of identification tags for each pattern, which is simply a set of unit vectors, uniquely tagging each tone (percept). The mapping of response patterns to unit vectors can be thought of as forming optimal tonotopic maps, given the coarse coding of the auditory periphery.

Through conjoint association, a matrix M forms as a function of Hebbian learning[3]. The derived matrix, which can now be thought of as a fuzzy relational instantiation of emergent implication, is shown in Figure 6. The surface described by the fuzzy relation corresponds to the connection strengths/compatibility values between the linguistic variable quantified by the response areas and the output auditory percepts.

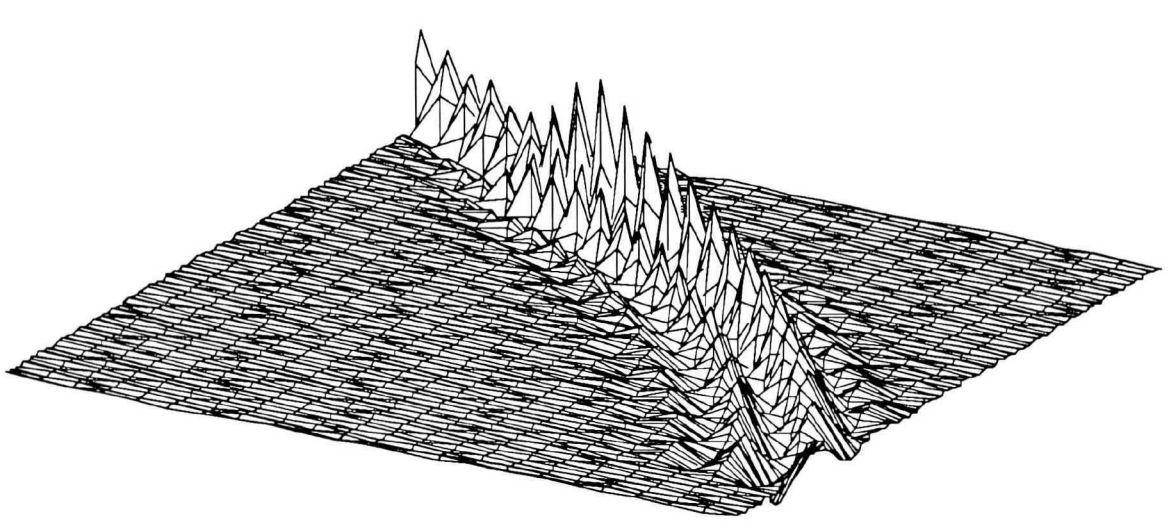


Fig. 6. Fuzzy relation derived from response function vector inputs and output identity matrix.

This exposition of implication in the auditory neural substrate has been strictly illustrative, but it is interesting to consider what this system might "infer" given some acoustic input. That is, what perceptual event will be taken to be implied by given sensory facts? Consider the composite input response function shown in Figure 7, representing the excitation pattern of three tones coded by the response areas in Figure 5.

[3]To achieve the desired independence of coding, one needs to augment a simple Hebbian feedforward net with mutual inhibition at the output nodes (Matsuoka, 1989). In our example, this was accomplished implicitly by means of the application of the pseudoinverse operator (see footnote 2 and the reference therein).

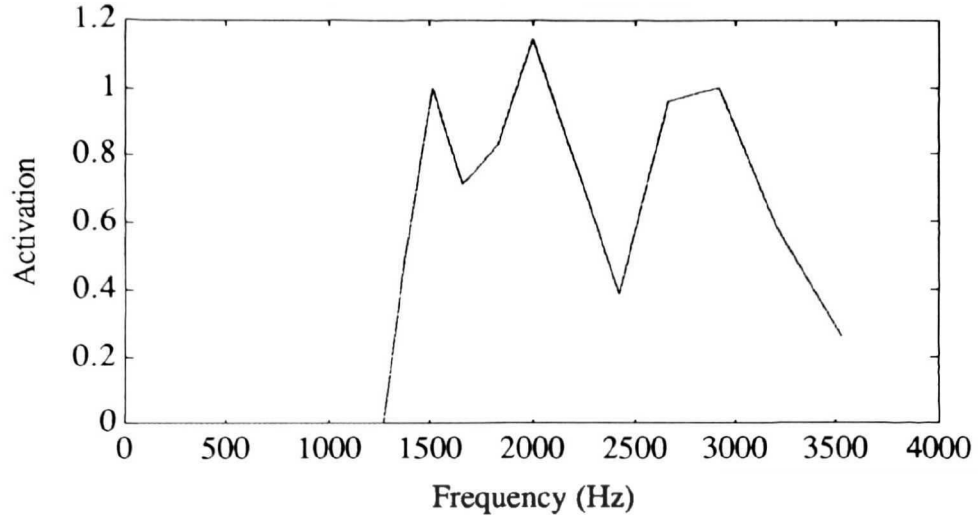


Fig. 7. Excitation pattern generated by 1500, 2000, and 2800 Hz.

The vector for this combination is the value assigned to the linguistic variable A. Now we apply the matrix multiplication form of the composition operator to A together with the fuzzy relation (emergent implication) matrix R shown in Figure 6 to infer some output B, shown in Figure 8. We observe that although the implied output from the linguistic variable A and the fuzzy relation has similarities to its antecedent, such as three prominent peaks, it exhibits certain interesting differences as well. The three frequency components are clearly more distinctive than initially suggested by the coarse coded representation .

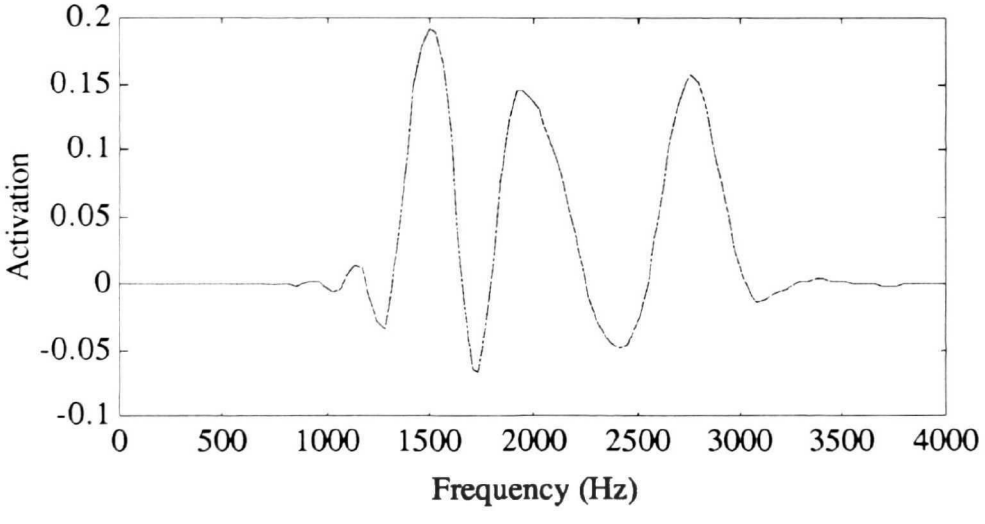


Fig. 8. Implied output vector from compositional inference.

The effect of compositional inference is one of integrating information that is coarse coded to produce an optimal output. In addition, the implied output exhibits a degree of lateral inhibition that suppresses neighboring percepts, as shown by the negative values in Figure 8. This phenomenon has often been described in the auditory system as well as other sensory systems (Sachs and Kiang, 1968) Thus, the implied percept may indeed have perceptual reality.

Conclusion

We have developed a simple account of emergent implication from unsupervised exposure to linguistic variables. The concepts used in constructing the framework of emergent implication were taken from Zadeh's notion of implication as a fuzzy relation and from connectionist models of associative memory. An application of this machinery to the peripheral auditory system illustrated how it may serve as a level of symbolic description linked by means of emergent implication to a subsymbolic description thereby achieving the kind of bistratal account advocated by Oden (1988).

References

Jenison, R. L., Kluender, K. R., Greenberg, S., and Rhode, W. S. (1990). *Analytic derivation of filter response functions from auditory nerve fiber data.* Paper presented at the 13th annual meeting of the Association for Research in Otolaryngology, St. Petersburg, Florida.

Kohonen, T. (1987). *Self organization and associative memory.* Berlin: Springer-Verlag.

Matsouka, K. (1989). An associative network with cross inhibitory connections. *Biological Cybernetics*, 61, 393-399.

Peterson, T.L., and Boll, S.F. (1983). Critical band analysis-synthesis. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 31, 656-663.

Oden, G. C. (1984). Integration of fuzzy linguistic information in language comprehension. *Fuzzy Sets and Systems*, 14, 29-41.

Oden, G. C. (1988). FuzzyProp: A symbolic superstrate for connectionist models. *Proceedings of the Second IEEE International Conference on Neural Networks, vol. 1*, 293-300.

Sacks, M.B., and Kiang, N.Y. S. (1968). Two-tone inhibition in auditory nerve fibers. *Journal of the Acoustical Society of America*, 43, 1120-1128.

Zadeh, L.A. (1975). The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences*, 8, 199-249; 8, 301-357; 9, 43-80.

Address correspondence to Rick Jenison, Department of Psychology, University of Wisconsin, Madison WI 53706 (jenison@vms.macc.wisc.edu). Gregg Oden is now at the University of Iowa.

Scenes from Exclusive-Or: Back Propagation is Sensitive to Initial Conditions

John F. Kolen & Jordan B. Pollack
Laboratory for Artificial Intelligence Research
Computer and Information Science Department
The Ohio State University
Columbus, Ohio 43210, USA
kolen-j@cis.ohio-state.edu, pollack@cis.ohio-state.edu

This paper explores the effect of initial weight selection on feed-forward networks learning simple functions with the back-propagation technique. We first demonstrate, through the use of Monte Carlo techniques, that the magnitude of the initial condition vector (in weight space) is a very significant parameter in convergence time variability. In order to further understand this result, additional deterministic experiments were performed. The results of these experiments demonstrate the extreme sensitivity of back propagation to initial weight configuration.

Back Propagation (Rumelhart, Hinton, & Williams, 1986) is the network training method of choice for many cognitive modeling projects, and for good reason. Like other weak methods, it is simple to implement, faster than many other "general" approaches, well-tested by the field, and easy to mold (with domain knowledge encoded in the learning environment) into very specific and efficient algorithms.

Rumelhart et al. made a confident statement: for many tasks, "the network rarely gets stuck in poor local minima that are significantly worse than the global minima."(p. 536) According to them, initial weights of exactly 0 cannot be used, since symmetries in the environment are not sufficient to break symmetries in initial weights. Since their paper was published, the convention in the field has been to choose initial weights with a uniform distribution between plus and minus ρ, usually set to 0.5 or less.

The convergence claim was based solely upon their empirical experience with the back propagation technique. Since then, Minsky & Papert (1988) have argued that there exists no proof of convergence for the technique, and several researchers (Judd 1988; Blum and Rivest 1988; Kolen 1988) have found that the convergence time must be related to the difficulty of the problem, otherwise an unsolved computer science question ($P \stackrel{?}{=} NP$) would finally be answered. We do not wish to make claims about convergence of the technique in the limit (with vanishing step-size), or the relationship between task and performance, but wish to talk about a pervasive behavior of the technique which has gone unnoticed for several years: the sensitivity of back propagation to initial conditions.

Initially, we performed empirical studies to determine the effect of learning rate, momentum rate, and the range of initial weights on t-convergence (Kolen and Goel, 1989). We use the term *t–convergence* to refer to whether or not a network, starting at a precise initial configuration, could learn to separate the input patterns (correct outputs above or below .5) within t epochs. The experiment consisted of training a 2-2-1 network on exclusive-or while varying three independent variables in 114 combinations: learning rate, η, equal to 1.0 or 2.0; momentum rate, α, equal to 0.0, 0.5, or 0.9; and initial weight range, ρ, equal to 0.1 to 0.9 in 0.1 increments, and 1.0 to 10.0 in 1.0 increments. Each combination of parameters was used to initialize and train a number of networks.[1] Figure 1 plots the percentage of t-convergent (where t = 50,000 epochs of 4 presentations) initial conditions for the 2-2-1 network trained on

[1]Numbers ranged from 8 to 8355, depending on availability of computational resources. Those data points calculated with small samples were usually surrounded by data points with larger samples.

the exclusive-or problem. From the figure we thus conclude the choice of $\rho \leq 0.5$ is more than a convenient symmetry-breaking default, but is quite necessary to obtain low levels of nonconvergent behavior.

Why do networks exhibit the behavior illustrated in Figure 1? While some might argue that very high initial weights (i.e. $\rho > 10$) lead to very long convergence times since the derivative of the semi-linear sigmoid function is effectively zero for large weights, this does not explain the fact that when ρ is between 2 and 4, the non-t-convergence rate varies from 5 to 50 percent.

Thus, we decided to utilize a more deterministic approach for eliciting the structure of initial conditions giving rise to t-convergence. Unfortunately, most networks have many weights, and thus many dimensions in initial-condition space. We can, however, examine 2-dimensional slices through the space in great detail. A slice is specified by an origin and two orthogonal directions (the X and Y axes). In the figures below, we vary the initial weights regularly throughout the plane formed by the axes (with the origin in the lower left-hand corner) and collect the results of running back-propagation to a particular time limit for each initial condition. The map is displayed with grey-level linearly related to time of convergence: black meaning not t-convergent and white representing the fastest convergence time in the picture. Figure 2 is a schematic representation of the networks used in this and the following experiment. The numbers on the links and in the nodes will be used for identification purposes. Figures 3 through 11 show several interesting "slices" of the the initial condition space for 2-2-1 networks trained on **exclusive-or**. Each slice is compactly identified by its 9-dimensional weight vector and associated learning/momentum rates. For instance, the vector (-3+2+7-4X+5-2-6Y) describes a network with an initial weight of -0.3 between the left hidden unit and the left input unit. Likewise, "+5" in the sixth position represents an initial bias of 0.5 to the right hidden unit. The letters "X" and "Y" indicate that the corresponding weight is varied along the X- or Y-axis from -10.0 to +10.0 in steps of 0.1. All the figures in this paper contain the results of 40,000 runs of back-propagation (*i.e.* 200 pixels by 200 pixels) for up to 200 epochs (where an epoch consists of 4 training examples).

Figures 12 and 13 present a closer look at the sensitivity of back-propagation to initial conditions. These figures zoom into a complex region of Figure 11; the captions list the location of the origin and step size used to generate each picture.

Sensitivity behavior can also be demonstrated with even simpler functions. Take the case of a 2-2-1 network learning the **or** function. Figure 14 shows the effect of learning "or" on networks (+5+5-1X+5-1Y+3-1) and varying weights 4 (X-axis) and 7 (Y-axis) from -20.0 to 20.0 in steps of 0.2. Figure 15 shows the same region, except that it partitions the display according to equivalent solution networks after t-convergence (200 epoch limit), rather than the time to convergence. Two networks are considered equivalent[2] if their weights have the same sign. Since there are 9 weights, there are 512 (2^9) possible network equivalence classes. Figures 16 through 25 show successive zooms into the central swirl identified by the XY coordinate of the lower-left corner and pixel step size. After 200 iterations, the resulting networks could be partitioned into 37 (both convergent and nonconvergent) classes. Obviously, the smooth behavior of the t-convergence plots can be deceiving, since two initial conditions, arbitrarily alike, can obtain quite different final network configuration.

Note the triangles appearing in Figures 19, 21, 23 and the mosaic in Figure 25 corresponding to the area which did not converge in 200 iterations in Figure 24. The triangular

[2]For rendering purposes only. It is extremely difficult to know precisely the equivalence classes of solutions, so we approximated.

boundaries are similar to fractal structures generated under iterated function systems (Barnsley 1988): in this case, the iterated function is the back propagation learning method. We propose that these fractal-like boundaries arise in back-propagation due to the existence of multiple solutions (attractors), the non-zero learning parameters, and the non-linear deterministic nature of the gradient descent approach. When more than one hidden unit is utilized, or when an environment has internal symmetry or is very underconstrained, then there will be multiple attractors corresponding to the large number of hidden-unit permutations which form equivalence classes of functionality. As the number of solutions available to the gradient descent method increases, the more complicated the non-local interactions between them. This explains the puzzling result that several researchers have noted, that as more hidden units are added, instead of speeding up, back-propagation slows down (e.g. Lippman and Gold, 1987). Rather than a hill-climbing metaphor with local peaks to get stuck on, we should instead think of a many-body metaphor: The existence of many bodies does not imply that a particle will take a simple path to land on one. From this view, we see that Rumelhart et al.'s claim of back-propagation usually converging is due to a very tight focus inside the "eye of the storm".

The emergence of chaotic phenomena in neural network models was certainly anticipated by both Kurten (1989) and Huberman (1987), carefully avoided (through the choice of symmetric weights) by Hopfield (1982), but usually disregarded by connectionists, including Rumelhart et al., despite the common knowledge that non-linearity is what enables these models to perform non-trivial computations in the first place.

What does this mean to the back-propagation community? From an engineering applications standpoint, where only the solution matters, nothing at all. From a cognitive science standpoint, when making claims that a back-propagation learning experiment is relevant to psychological data, **the initial conditions for the network need to be precisely specified or made publicly available.**

What about the future of back-propagation? We remain neutral on the issue of its ultimate convergence, but our result points to a few directions for improved methods. Since the slowdown occurs as a result of global influences of multiple solutions, an algorithm for first factoring the symmetry out of both network and training environment (e.g. domain knowledge) may be helpful. Furthermore, it may also turn out that search methods which harness "strange attractors" ergodically guaranteed to come arbitrarily close to some subset of solutions might work better than methods based on strict gradient descent. Thus, rather than abandoning back-propagation, we view this result as yet further impetus to discover how to exploit the information-creative aspects of non-linear dynamical systems for future models of cognition (Pollack 1989).

Acknowledgments

This work was partially supported by the Office of Naval Research. Substantial free use of over 200 Sun workstations was generously provided by our department.

References

M. Barnsley. 1988. *Fractals Everywhere*, Academic Press, San Diego, CA. 1988.

A. Blum and R. Rivest. 1988. Training a 3-node Neural Network is NP-Complete. *Proceedings of IEEE Conference on Neural Information Processing Systems*, Denver, Colorado, 1988.

J. J. Hopfield. 1982. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. *Proceedings US National Academy of Science* **79**:2554-2558.

B. A. Huberman and T. Hogg. 1987. Phase Transitions and Artificial Intelligence. *Artificial Intelligence*, **33**, 155-172.

S. Judd. 1988. Learning in Networks is Hard. *Journal of Complexity* **4**:177-192.

J. Kolen. 1988. Faster Learning Through a Probabilistic Approximation Algorithm. *Proceedings of the Second IEEE International Conference on Neural Networks*, San Diego, California, pp. I:449-454.

J. Kolen and A. Goel. 1989. Learning in Parallel Distributed Processing Networks: Computational Complexity and Information Content. (Tech. Rep., 89-JK-LEARNING). Columbus, Ohio: The Ohio State University, Laboratory for AI Research.

K. E. Kurten and J. W. Clark. 1986. Chaos in Neural Networks. *Physics Letters*, **114A**, 413-418.

R. P. Lippman and B. Gold. 1987. Neural Classifiers Useful for Speech Recognition. In *1st International Conference on Neural Networks* ,IEEE, pp. IV:417-426.

M. L. Minsky and S. A. Papert. 1988. *Perceptrons*. Cambridge, MA: MIT Press.

J. B. Pollack. 1989. Implications of Recursive Auto Associative Memories. In *Advances in Neural Information Processing Systems*. (ed. D. Touretzky) pp 527-536, Morgan Kaufman, San Mateo.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. 1986. Learning Representation by Back-Propagating Errors. *Nature* **323**:533-536.

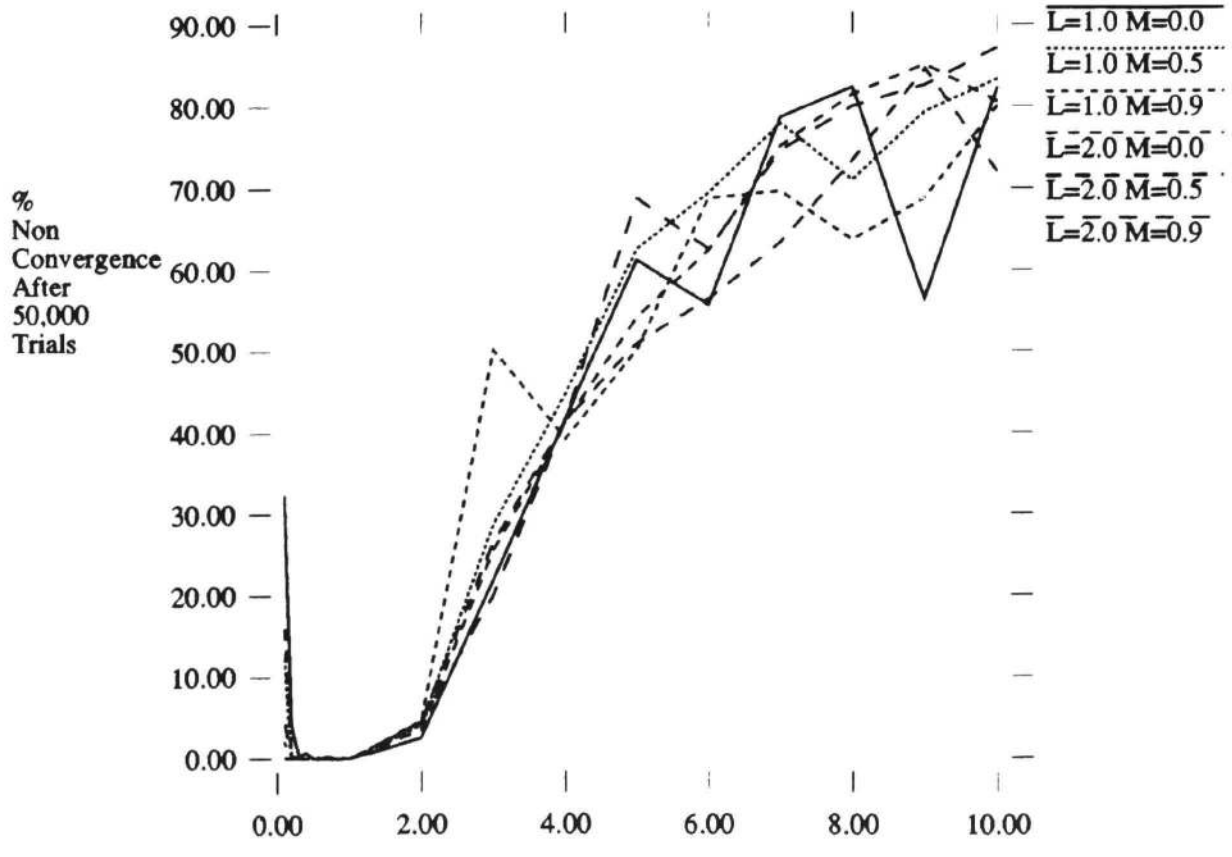


Figure 1: Percentage T-Convergence vs. Initial Weight Range

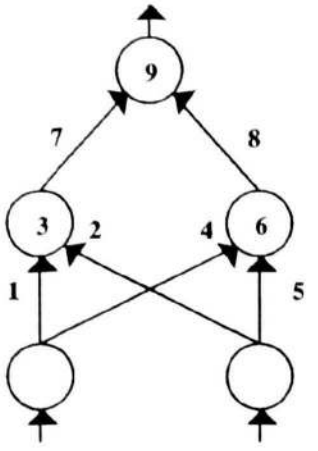


Figure 2 : Schematic Network

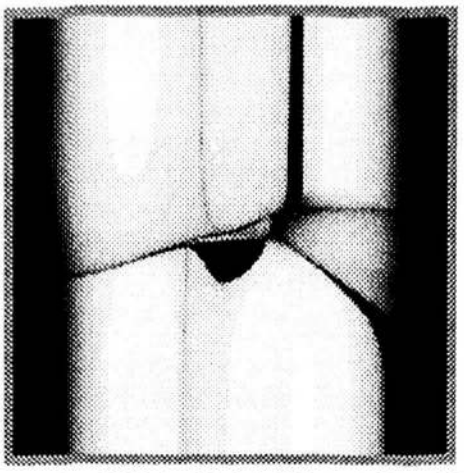


Figure 3 : (-5-3+3+6Y-1-6+7X) η=3.25 α=0.40



Figure 4 : (+4-7+6+0-3Y+1X+1) η=2.75 α=0.00



Figure 5 : (-5+5+1-6+3XY+8+3) η=2.75 α=0.80

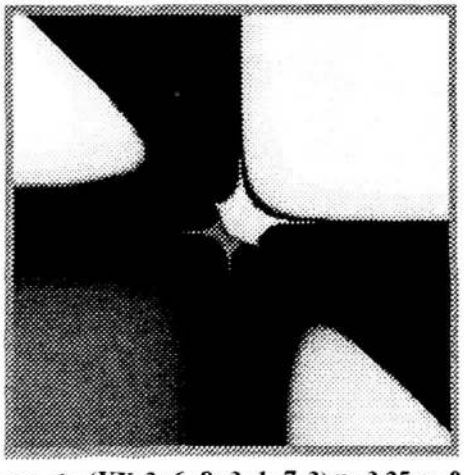


Figure 6 : (YX-3+6+8+3+1+7-3) η=3.25 α=0.00

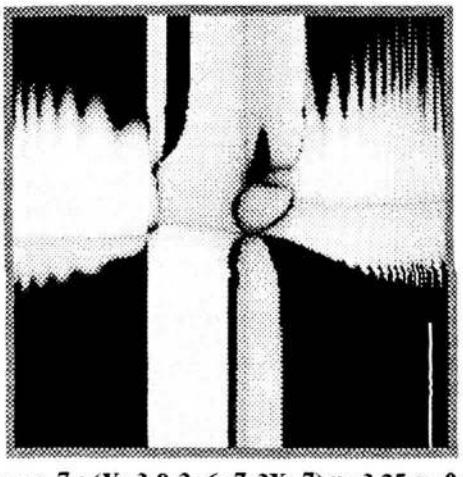


Figure 7 : (Y+3-9-2+6+7-3X+7) η=3.25 α=0.60

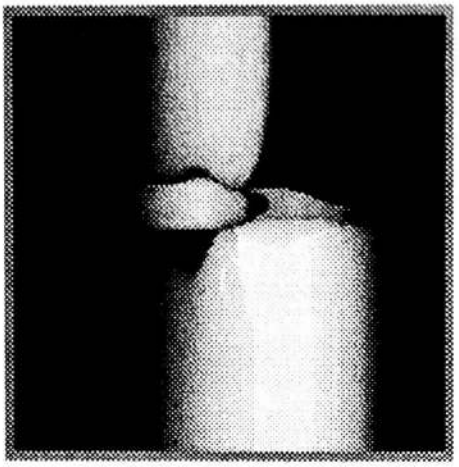


Figure 8 : (-6-4XY-6-6+9-4-9) η=3.00 α=0.50

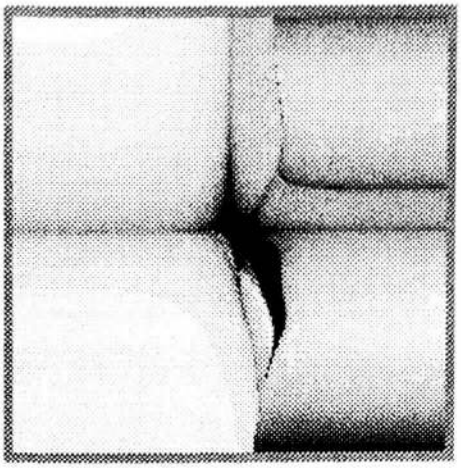


Figure 9 : (-2+1+9-1X-3+8Y-4) η=2.75 α=0.20

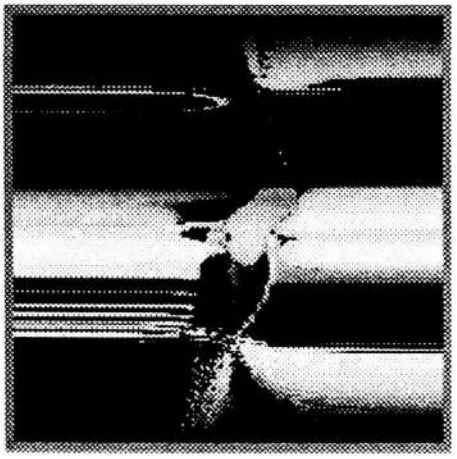


Figure 10 : (+1+8-3-6X-1+1+8Y) η=3.50 α=0.90

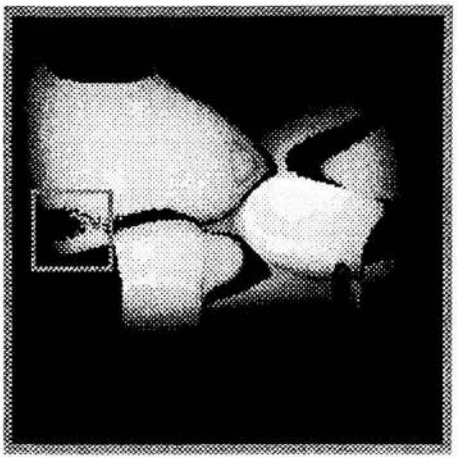


Figure 11 : (+7+4-9-9-5Y-3+9X) η=3.00 α=0.70

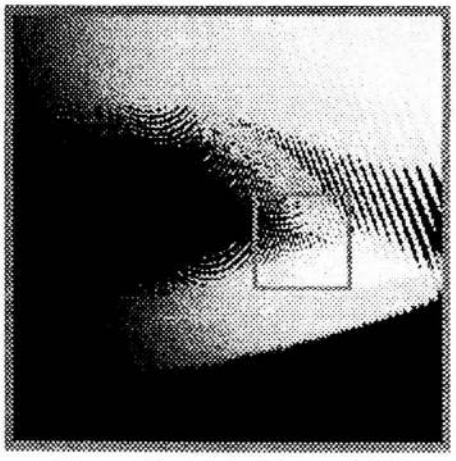


Figure 12 : (-9.0,-1.8) step 0.018

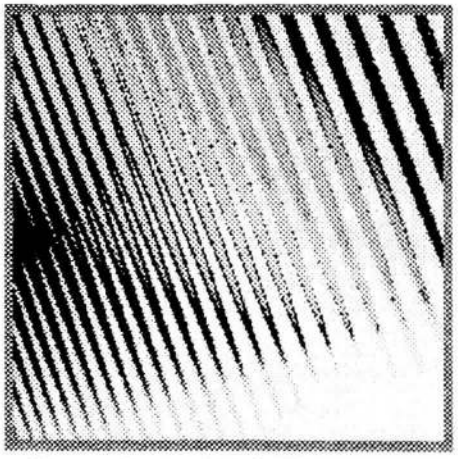


Figure 13 : (-6.966,-0.500) step 0.004

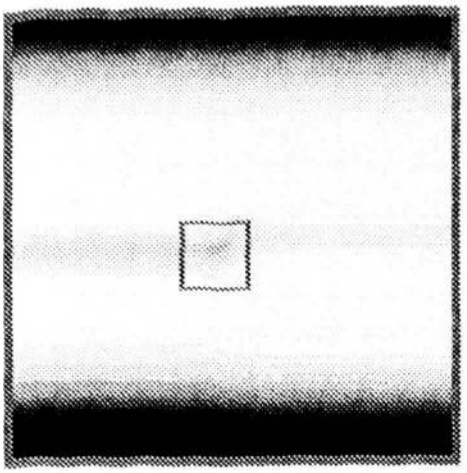


Figure 14 : (-20.00000, -20.00000) step 0.200000

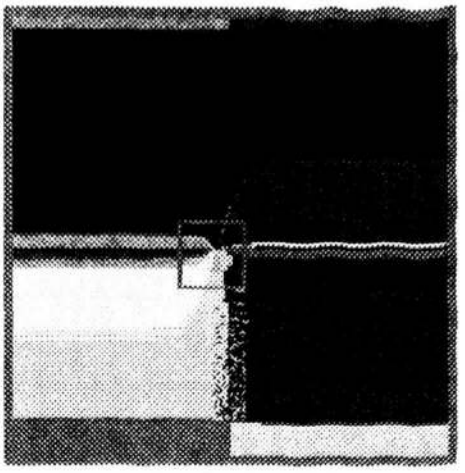


Figure 15 : Solution Networks

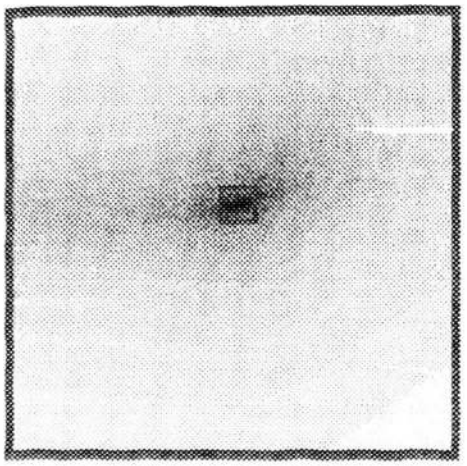


Figure 16 : (-4.500000, -4.500000) step 0.030000

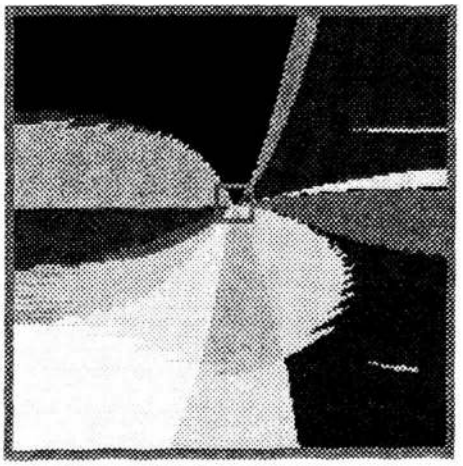


Figure 17 : Solution Networks

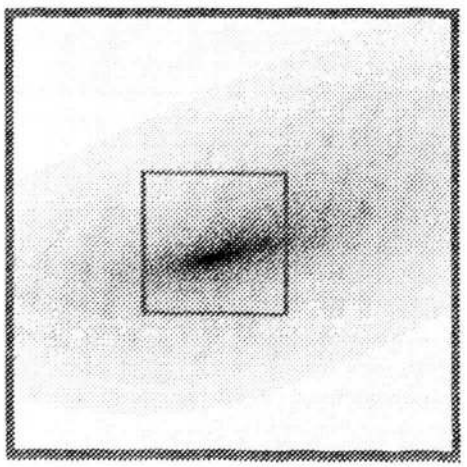


Figure 18 : (-1.680000, -1.350000) step 0.002400

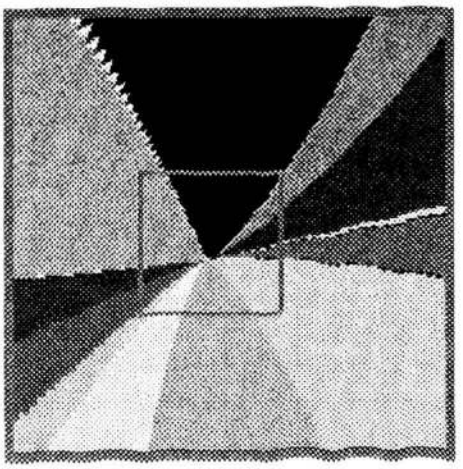


Figure 19 : Solution Networks



Figure 20 : (-1.536000, -1.197000) step 0.000780

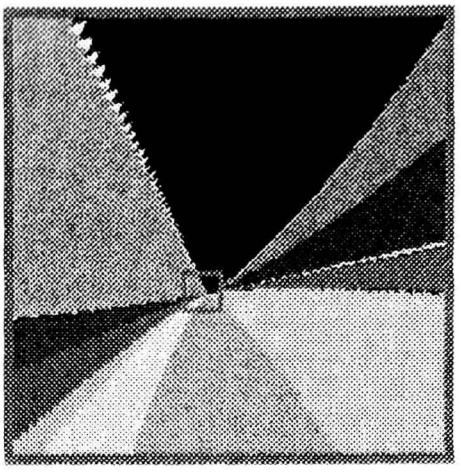


Figure 21 : Solution Networks

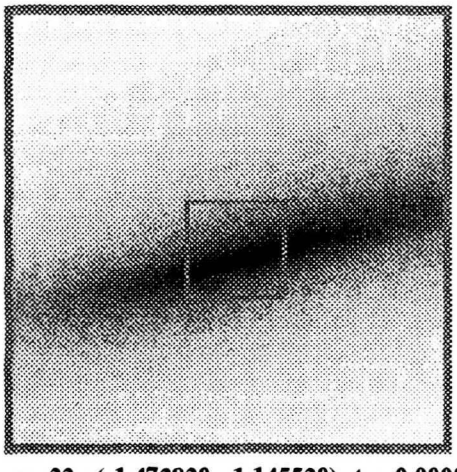


Figure 22 : (-1.472820, -1.145520) step 0.000070

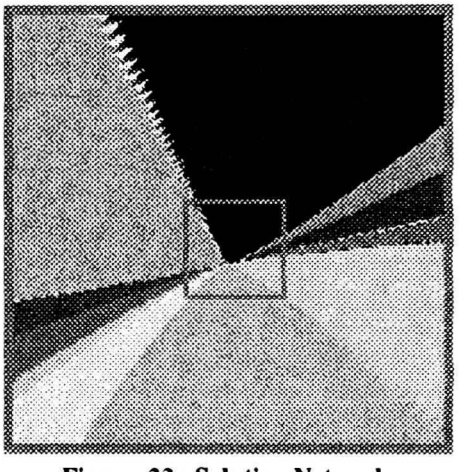


Figure 23 : Solution Networks

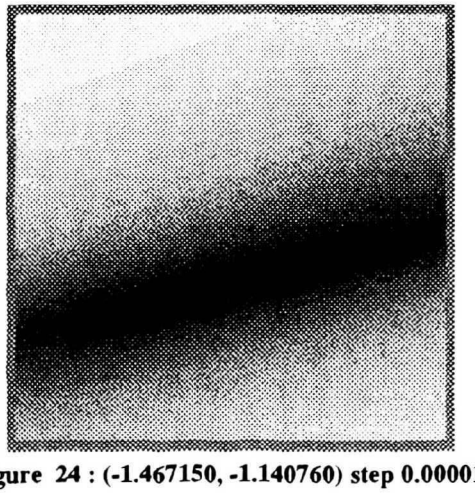


Figure 24 : (-1.467150, -1.140760) step 0.000016

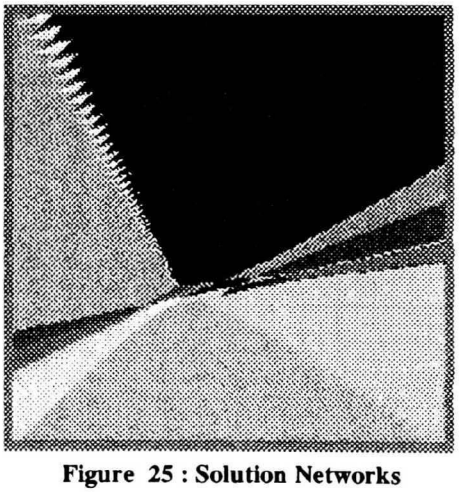


Figure 25 : Solution Networks

A COMPUTATIONAL MODEL OF ATTENTIONAL REQUIREMENTS IN SEQUENCE LEARNING

Peggy J. Jennings and Steven W. Keele
Department of Psychology
University of Oregon

ABSTRACT

This paper presents a computational model of attentional requirements in sequence learning. The structure of a keypressing sequence affects subjects' abilities to learn the sequence in a dual task paradigm (Cohen, Ivry, & Keele, 1990). Sequences containing unique associations among successive positions (i.e., 1-5-4-2-3) are learned during distraction. Sequences containing repeated positions with ambiguous associations (i.e., 3-1-2-1-3-2) are not learned during distraction. Cohen, et al. proposed two fundamental operations in sequence learning. An associative mechanism mediates learning of the unique patterns (1-5-4-2-3). These associations do not require attention to be learned. Such an associative mechanism is poorly suited for learning the sequence with repeated elements and ambiguous associations. These sequences must be parsed and organized in a hierarchical manner. This hierarchical organization requires attention.

The simulations reported in this paper were run on an associative model of sequence learning developed by Jordan (1986). Sequences of differing structures were presented to the model under two conditions -- unparsed, and parsed into subsequences. The simulations modeled closely the keypressing task used by Cohen, Ivry and Keele (1990). The simulations (1) replicate the empirical findings, and (2) suggest that imposing hierarchical organization on sequences with ambiguous associations significantly improves the model's ability to learn those sequences. Implications for the analysis of fundamental computations underlying a system of skilled movement are discussed.

This paper examines whether a connectionist model of sequence learning developed by Jordan (1986, 1990) provides insight into the attentional requirements of sequence learning as investigated by Cohen, Ivry, and Keele (1990; see also Keele, Cohen, & Ivry, 1990). The first section describes the central phenomena explored by Cohen, et al. The second section describes aspects of Jordan's model and how it relates to the empirically discovered phenomena.

BEHAVIORAL CHARACTERISTICS OF SEQUENCE LEARNING

Cohen, et al. adopted a paradigm established by Nissen and Bullemer (1987) to investigate the learning of sequential representations. Successive presentations of a visual stimulus appeared at one of 3, 4, or 5 locations on a screen. Subjects responded to each stimulus by pressing a key corresponding to the stimulus location. Reaction times were recorded. Unknown to the subjects, a large proportion of the successive signals appeared in a particular repeating sequence of locations. One type of sequence, called Unique, involves a sequence of 5 unique signal positions, an example of which is 1-5-4-2-3. The numbers refer to signal positions. After the last position in a cycle, the sequence repeated with no detectable break. Sequences were presented to subjects in 8 blocks of 20 cycles through the sequence. A second type of sequence, called Ambiguous, involves only 3 signal positions. Each position is repeated within the sequence, but each occurrence is followed by a different successor. An example of an Ambiguous sequence is 1-3-2-3-1-2.

Cohen, et al. found that, with practice, subjects learn both of these types of sequences in the absence of a distraction task. Sequence acquisition was demonstrated by steady improvement in reaction time over 8 blocks of training, with a significant increase in reaction time on blocks of trials presented after training in which the signals occurred at random rather than in the structured sequence. When a secondary task was performed simultaneously, diverting attention from the primary reaction time task, Unique sequences were learned but Ambiguous ones were not. That is, performance to signals occurring in the structured Ambiguous sequence never became faster than performance to randomly occurring signals.

Something about learning the Ambiguous sequence appears to require attention. Cohen, et al. also observed that if an Ambiguous sequence is altered by replacing one of the repeated events with an event that occurs only once in the cycle, such as 1-4-2-3-1-2 (the underline shows the unique event), subjects learn this sequence at a rate similar to that for the Unique sequences. These sequences are called Hybrid because they involve a mixture of repeated and unique events.

Cohen, et al. offered the following explanation of the effects of sequence structure on learning. The Unique sequences can be acquired by forming associations between adjacent events. Such learning appears to require little or no attention to the relationships among items. Associational learning is not well suited for learning Ambiguous sequences, however, because on different occasions a particular event is ambiguously followed by a different event. Ambiguous sequences can, however, be learned by a mechanism that divides the sequence into parts. This process of hierarchic organization seems to require attention. Why the Hybrid sequences are learned more like Unique than Ambiguous sequences is unclear.

A COMPUTATIONAL MODEL OF SEQUENCE LEARNING

The goal of the present research is to gain insight into (1) how the structure of a sequence interacts with component operations of a sequence learning system; and (2) how parsing and hierarchic representation may be implemented in a neural network. These goals were approached using Jordan's (1986, 1990) recurrent network model. His model, as we implemented it, is illustrated in Figure 1. An input layer has units of two types: plan units and context units. These input units are connected to a layer of hidden units, and those hidden units feed into output units that we call prediction units. Activation of these latter units represents a prediction, or priming, of the upcoming event. The prediction units can be viewed variously as representing a prediction of the upcoming stimulus or, because the stimulus determines which response to make, the upcoming response. Stimulus information constitutes the target output that is used to calculate error in the prediction units and forms the basis for

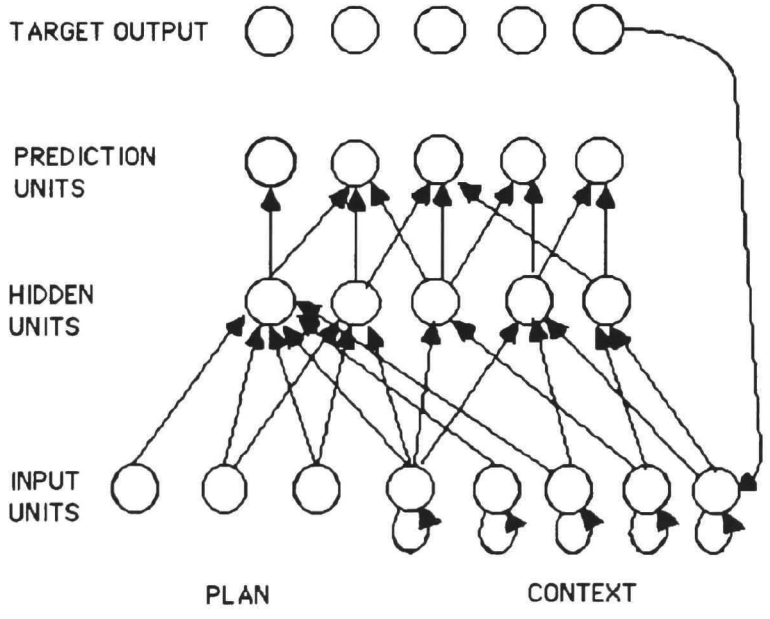


Figure 1. Jordan (1986) recurrent network model. (Not all connections are shown.)

learning using backpropagation. Changes in connection weights are made over the course of learning based on the amount of prediction error on each trial.

During training, information about the target response feeds back to the context units with a fixed connection weight of 1. Output representation is local rather than distributed, and each target

output unit is connected to one and only one context unit. Each context unit also feeds a proportion of its previous activation back on itself. The recurrent connections produce context unit activations that retain a history of recent events. In an Ambiguous sequence such as 1-3-2-3-1-2, the context following stimulus 3 is somewhat different each time the 3 occurs because in each case it is preceded by different events. The influence of those events, while diminishing over successive stimuli, is partially retained by their representation in the context units.

Sequence learning is achieved as follows: at the beginning of a sequence a pattern of activation appears on the plan units. In the general case, this pattern persists in unmodified form throughout the training session. At the beginning of a block of training, the context units are all set to zero. The combination of plan and initial context values feed through the hidden layer to produce a prediction. A "stimulus", or target response, provides target output information. Prediction error is calculated, and weight changes occur via backpropagation. The "stimulus" information also feeds back to the context units, and the context units reverberate a portion of their previous values. On subsequent iterations, the updated context values in combination with the plan values constitute the input patterns associated with consecutive target responses.

In these simulations, the model was trained on two distinct types of sequence organization. One condition involved presenting 20 cycles through each sequence type with no higher-level organization. In another condition, the input patterns were parsed into distinct subparts. Two implementations of parsing were examined. One implementation involved resetting the context units to zero at the beginning of each cycle through the sequence. This implements a form of parsing whereby the boundaries of a sequence are marked. The second implementation of parsing involved assigning different plan values to each subpart of a sequence. When a subsequence is finished and a new one is to start, the activation pattern on the plan units changes accordingly. In each implementation, parsing is made explicit in the input patterns.

The intent of these simulations is to determine how parsing influences the learning of the three sequence types: Unique, Hybrid, and Ambiguous. Parsing, in this conception, corresponds in the empirical case to an attention-dependent organizational process that mediates learning of Ambiguous sequences.

Simulation 1 -- Associative Processes in Sequence Learning

The goal of Simulation 1 was to replicate the behavioral characteristics of sequence learning under divided attention as reported by Cohen, Ivry, and Keele (1990). Under dual-task conditions, performance of Unique sequences (e.g., 1-5-4-2-3) and Hybrid sequences (e.g., 1-4-2-3-1-2) steadily improves with training, while performance of Ambiguous sequences (e.g., 1-3-2-3-1-2) remains no better than performance to randomly presented stimuli.

Training Set

Input to the model was composed of 2 elements. (1) a plan value; and (2) a representation of recent target keypress history. Plan values remained constant throughout the training session for each sequence. Target keypress values were represented by a vector of binary values across 5 units. For example, an output pattern of 00100 represents a press of the middle key. Context values represented the sum of the desired output at time t and a proportion of the context value at time t-1. Training sets consisted of 20 cycles through a sequence, consistent with a single block of training in the empirical task. Each input pattern is associated with a desired output value corresponding to a keypress. At the beginning of a block of training, context units were set to zero. Performance measures consisted of (1) total sum of squares of prediction error summed over individual trials and recorded after each block of training; and (2) number of blocks required to learn the sequence to specified accuracy criterion (less than 0.04 sum of squares prediction error over 20 cycles). The model was trained on three exemplars of each sequence type: Unique (1-5-2-4-3; 1-4-5-3-2; 1-3-4-2-5), Ambiguous (1-2-3-1-3-2; 1-2-3-2-1-3; 1-3-2-3-1-2), and a Hybrid of unique and ambiguous associations (1-2-3-1-3-4; 1-2-3-2-4-3; 1-4-2-3-1-2). Six separate training sessions per sequence type were run; two sessions per sequence. Mean values from sequence type groups were analyzed.

Results of Simulation 1

An analysis of variance of number of blocks required to learn the sequence to criterion showed a significant effect of sequence type ($F(2,15) = 19.742$, $p <= 0.001$). Multiple comparisons (Tukey, $p < .05$) reveal that the model requires fewer blocks of training to learn the Unique sequences (mean blocks = 62.2) and the Hybrid sequences (mean blocks = 68.0) than to learn the Ambiguous sequences (mean blocks = 106.0). The Unique and Hybrid conditions are not significantly different.

Figure 2 displays the rate of change in prediction error over the first 10 blocks of training. While learning the Unique sequences, the reduction in prediction error is rapid and steadily decreasing with training, approaching zero. While learning the Amibiguous sequences, however, prediction error levels off at a relatively high value after an initial brief and rapid decline. These error measures may be compared with improvement in reaction time reported in the empirical case, resulting from the subject's increasing ability to correctly anticipate events. In the Ambiguous condition, prediction error as well as empirical reaction times remain relatively high as anticipation of response is not much better than chance.

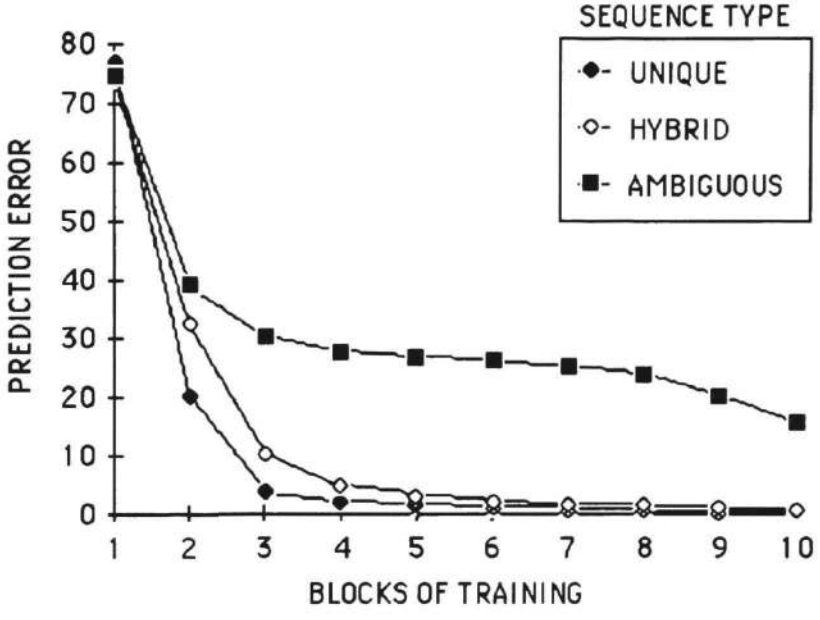


Figure 2. Prediction error as a function of training, by sequence type.

Discussion

Learning in the simulation is denoted by reductions in error between prediction values and target response values while learning in humans is documented by reductions in reaction time. Human reaction time is improved by priming or expectancy. Model error as well as empirical reaction times, because block means are reported, reflect accuracy of prediction over the course of 20 cycles through a sequence.

In this simulation, both Unique and Hybrid sequences were learned at essentially the same rate. These results agree qualitatively with the results of human learning in which learning of Unique and Hybrid sequences also occurs at similar rates when a distraction task is used. In contrast to the equivalence of Unique and Hybrid sequence learning rates, the model learns the Ambiguous sequences more slowly. This result is qualitatively similar to the empirical result in which a distraction task prevents learning of the Ambiguous sequences. It must be noted, however, that Ambiguous sequence learning does occur in the simulation. For the Unique and Hybrid sequences, the model required over 50 blocks to learn the sequences, while human subjects show learning within 8 blocks of training. It is not clear how the time frames between the model and the human are related, but given that the model re-

quired 106 blocks to learn the Ambiguous sequence, it is possible that humans may also show evidence of learning the Ambiguous sequences with more extensive practice.

Simulation 2 -- Hierarchical Organization of Input Sequences

Cohen, et al. proposed that the role of attention in sequence learning is to mediate parsing of sequences containing ambiguous associations. The goal of Simulation 2 was to examine the effects on learning of imposing an hierarchical organization on the input sequence. The associative learning mechanism implemented in Simulation 1 required significantly more training (106 blocks) to learn the Ambiguous sequences than to learn the Unique and the Hybrid sequences (65 blocks). Preorganizing the input patterns into parsed subsequences should allow the associative mechanism to now learn the Ambiguous sequences as quickly as it learns Unique and Hybrid sequences. Because the associative mechanism is already efficient at learning Unique and Hybrid sequences, performance on them should remain virtually unchanged by delineating boundaries in the sequence patterns.

Training Set

The training sets in Simulation 2 were similar to those used in Simulation 1 except that, in Simulation 2, parsing was imposed on the input patterns. Parsing was achieved in two alternative ways. In one condition (Simulation 2.1), context unit activation was set to 0 at the beginning keypress of each cycle of the sequence within a block of 20 cycles. Unique, Hybrid, and Ambiguous sequences were parsed in this way and presented to the model. In this condition, the internal structure of the sequence is left intact, but the beginning of each cycle is delineated by a change in context. Here parsing is achieved by making explicit changes in activation values of the context units, while leaving plan unit values constant. In another condition (Simulation 2.2), which affected only the Ambiguous sequences, parsing was achieved by modifying the value of the plan units at the beginning of each subsequence within the Ambiguous sequence. For example, the sequence 3-1-2-1-3-2 had a plan value of 110 for the 3-1-2 subsequence, and 111 for the 1-3-2 subsequence. During training, therefore, the subplan values alternated as the sequence progressed. In this condition, the Ambiguous sequence is represented as two alternating subsequences. Here parsing is achieved by making explicit changes to the plan units, while leaving the context unit values dependent on model function.

Results of Simulation 2.1

In the first parsing condition, resetting context values to zero, the Ambiguous sequences were learned as quickly as were the Unique and Hybrid sequences (mean blocks 50.8, 60.7 and 57.0, respectively). The three sequence types are not significantly different ($F(2,15) = 1.483$, $p < 0.26$).

An analysis of variance of the number of blocks of training required to learn the sequences in both Simulation 1 (unparsed) and Simulation 2.1 (parsed, zero context) revealed significant effects of sequence type (Unique, Hybrid, or Ambiguous) ($F(2,30) = 7.991$, $p < 0.002$), sequence organization (unparsed vs. parsed) ($F(1,30) = 33.659$, $p < 0.000$), and a significant interaction of sequence type and organization ($F(2,30) = 18.088$, $p < 0.000$). Unique and Hybrid sequences are learned more quickly than Ambiguous sequences (mean blocks = 61.4, 62.5, and 78.4, respectively). Parsed sequences of all three types are learned more quickly than unparsed sequences (mean blocks = 56.2 and 78.7, respectively). The main effects of sequence type and organization, as well as the interaction, are strictly the result of improvement in learning rate for the Ambiguous sequences in the parsed condition over the rate in the unparsed condition. Learning rates for Unique and Hybrid sequences are unimproved by parsing.

Results from post hoc comparisons (Tukey, $p < .05$) of learning rates for the three sequence types in the parsed and unparsed conditions support the hypothesis that unparsed Unique and Hybrid sequences as well as parsed Unique, Hybrid, and Ambiguous sequences are all learned at the same rate (mean blocks = 59.74). Only unparsed Ambiguous sequences are significantly more slowly learned (mean blocks = 106.00). The groupings are as follows:

GROUP A: Unparsed (Unique, Hybrid); Parsed (Unique, Hybrid, Ambiguous)
GROUP B: Unparsed (Ambiguous)

Results of Simulation 2.2

In the second parsing condition, involving alternating subplans in Ambiguous sequences, the parsed sequences were learned in 55.3 blocks. An analysis of variance of blocks to learn the Ambiguous sequences under three types of organization (unparsed, parsed (zero context), and parsed (alternating subplans)) reveals a significant effect for organization ($F(2,15)=24.958$, $p<0.000$). Post hoc comparisons show that the parsed sequences are learned at the same rate, while the unparsed sequences require significantly more exposure to the sequence to learn it. In this case, the particular implementations of parsing are not significantly different. Figure 3 summarizes the results of Simulation 2, displaying the rate of improvement in model prediction as a function of training for parsed and unparsed Ambiguous sequences. The learning rate for both implementations of parsing is similar to that required to learn the parsed Unique and Hybrid sequences, while the learning rate for the unparsed Ambiguous sequence remains at a relatively high level of prediction error.

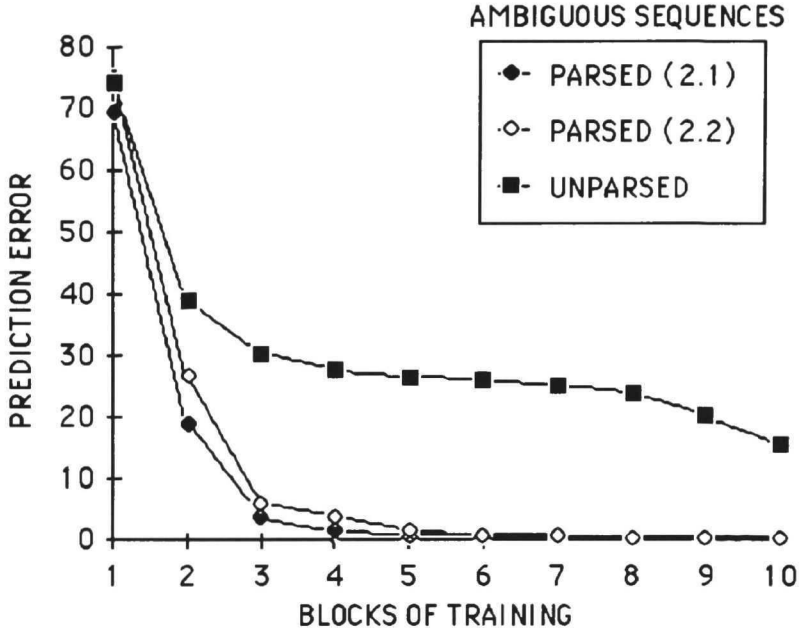


Figure 3. The effect of sequence organization on learning Ambiguous sequences

Discussion

Each particular method of implementing parsing had the same effect on sequence learning. When parsed in either of these two ways, Ambiguous sequences are learned by an associative mechanism at a rate similar to that required to learn Unique and Hybrid sequences. These results agree with the empirical results in which removal of distraction (presumably allowing an attention-dependent parsing process to function) enables, or at least greatly enhances, learning of the Ambiguous sequences.

GENERAL DISCUSSION

The two forms of parsing used in the simulation correspond to two different mechanisms, both of which could be involved in human sequence learning. One form of parsing involves resetting context units to zero at the beginning of each cycle of the sequence while leaving plan values intact. This form of parsing may be seen as mimicking a working memory function. The sequence is represented by activation across the plan units, but learning is facilitated by identifying the boundaries of the sequence. When a cycle of the sequence ends, plan values denote that the same sequence will follow, but the change in context values provides information about where the sequence begins and ends as well as indirect information about the length of the sequence.

The second form of parsing was one in which plan values were altered to represent sequence subparts, while leaving the context units to function uninterrupted. This notion of parsing is somewhat closer to traditional notions of hierarchic representation in which a node stands as a symbol for a more complex and lower order sequence of events. It may be seen as simulating the function of higher level object recognition processes. The alternating subplans represent the final products of a mechanism which has identified the subparts of a sequence whole.

The components of the Jordan model provide further insights into the various computations involved in learning complex motor sequences. The foundation for this learning system is the associative mechanism represented in the model by connections strengths between input patterns and response predictions. A simple two-layer system is fully capable of learning the Unique sequences in these simulations. Hidden units mediate higher level abstractions of input patterns. The hidden units perform an intermediate synthesis of input features. Context units function in a way that is analogous to visuospatial working memory. And finally, plan units reflect the contributions of representational or symbolic processes.

The model used here to simulate sequence learning represents the major components of a sequence learning system. Some of these component processes are dependent on attention, while others are not. It appears that various types of motor sequences require differing levels of interaction with these representation and memory components depending on the structure of the sequence. Unique sequences may require only a strict associative mechanism. Hybrid sequences may require association and synthesis as well as working memory capacity. Ambiguous sequences may need the full system of association, synthesis, working memory, and symbolic representation. These simulations have shown that learning of ambiguous associations are facilitated by higher level processes of organization. Facilitation of learning may occur at the level of working memory by providing cues to pattern boundaries. Facilitation may also occur as the result of an attention-dependent process which parses input sequences into subparts in a hierarchical representation. These effects of different parsing implementations provide insight into the reason that Hybrid sequences are learned by human subjects and by the simulation at a rate similar to that required to learn Unique sequences. The unique elements in the Hybrid sequence may serve to provide cues to boundaries within the sequence. These cues may function at the level of working memory and are independent of attention. This would allow Hybrid sequences to be learned during distraction, as the empirical data show. The Ambiguous sequences contain no clues to pattern boundaries, so they must be analyzed into subparts to be learned. This hierarchical organization process requires attention. The empirical data show that human subjects are able to learn the Ambiguous sequences when attention is not distracted to another task, but do not show evidence of learning the Ambiguous sequence under dual-task conditions. Future simulations will involve examining further the nature of these interactions between sequence structure, attention, and the fundamental operations that underly sequence learning.

ACKNOWLEDGMENTS

We are grateful for the financial and moral support from the Office of Naval Research (Contract N00014-87-K-0279) in the conduct of this research. We have benefitted greatly from discussions with Asher Cohen, Richard Ivry, and Clifford Keele.. We extend our thanks to Jay McClelland for making available a test version of Macintosh software on which these simulations were run. Discussions with Jim Tanaka during the early work on these simulations were helpful and very much appreciated. And thanks to Mike Posner for comments and discussions that are forever thought-provoking and insightful.

REFERENCES

Cohen, A., Ivry, R. I., & Keele, S. W. (1990). Attention and structure in sequence learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 16,* 17-30.

Jordan, M. I. (1986). *Serial order: A parallel distributed processing approach* (ICS Report 8604). La Jolla, CA: Institute for Cognitive Science, University of California, San Diego.

Jordan, M. I. (1990). Learning to articulate: Sequential networks and distal constraints. In M. Jeannerod (Ed.), Attention and performance XIII. Hillsdale, NJ: Erlbaum.

Keele, S. W., Cohen, A., & Ivry, R. I. (1990). Motor programs: Concepts and issues. In M. Jeannerod (Ed.), *Attention and performance XIII.* Hillsdale, NJ: Erlbaum.

Nissen, M. J., & Bullemer, P. (1987). Attentional requirements of learning: Evidence from performance measures. *Cognitive Psychology, 19,* 1-32.

Harmonic Grammar - A formal multi-level connectionist theory of linguistic well-formedness: An application

Géraldine Legendre[1,2], Yoshiro Miyata[1,3,4], Paul Smolensky[1,3,4]

[1]*Institute of Cognitive Science,* [2]*Department of Linguistics,* [3]*Department of Computer Science,*
& [4]*Optoelectronic Computing Systems Center, University of Colorado at Boulder*

Abstract

We describe *harmonic grammar*, a connectionist-based approach to formal theories of linguistic well-formedness. The general approach can be applied to various kinds of linguistic well-formedness, e.g., phonological and syntactic. Here, we address a syntactic problem: unaccusativity. Harmonic grammar is a two-level theory, involving a distributed, lower level connectionist network whose relevant aggregate computational behavior is described by a local, higher level network. The central hypothesis is that the connectionist well-formedness measure called "harmony"[1] can be used to model linguistic well-formedness; what is crucial about the relation between the lower and higher level networks is that there is a harmony-preserving mapping between them: they are *isoharmonic* (at least approximately). A companion paper (Legendre, Miyata, & Smolensky, 1990; henceforth "LMS_2") describes the theoretical basis for the two level approach, starting from general connectionist principles. In this paper, we discuss the problem of unaccusativity, give a high level characterization of harmonic syntax, and present a higher level network to account for unaccusativity data in French. We interpret this network as a fragment of the grammar and lexicon of French expressed in "soft rules." Of the 760 sentence types represented in our data, the network correctly predicts the acceptability in all but two cases. This coverage of real, problematic syntactic data greatly exceeds that of any other formal account of unaccusativity of which we are aware.

The Linguistic Problem: Unaccusativity

Since the goal of this work is to develop a genuine refinement of current symbolic linguistic accounts, we begin by devoting a few pages to a discussion of a linguistic problem and current symbolic approaches to it. Verbs have traditionally been characterized as either transitive or intransitive. Following a Relational Grammar proposal put forward by Perlmutter (1978) and adopted since by other grammar frameworks, e.g., Burzio (1986) in Government-Binding theory, formal syntacticians today generally agree that it is necessary to further divide intransitive verbs into two types, *unergatives* and *unaccusatives*, each associated with a different syntactic configuration. In Government-Binding theory, for example, the three classes of verbs differ with respect to their d(eep)-structure, as illustrated in (1).

(1) a. Transitive Verb: $[_S$ NP $[_{VP}$ V NP]]
b. Unergative Intransitive Verb: $[_S$ NP $[_{VP}$ V]]
c. Unaccusative Intransitive Verb: $[_S$ $[_{VP}$ V NP]]

While transitive verbs take two NP arguments, a subject and a direct object, unergative and unaccusative take only one: a deep subject (VP-external argument) in the case of unergatives, a deep direct object (VP-internal argument) in the case of unaccusative verbs.

The unaccusative/unergative distinction is motivated by the cross-linguistic fact that in certain syntactic contexts in which a verb may be embedded, one class of intransitive verbs are acceptable while the remaining are not. These syntactic environments are called *diagnostic contexts for unaccusativity*; we'll call them "diagnostic contexts" or simply "contexts." An example from French, Object Raising, is illustrated in (2).

(2) a. **La vérité** est facile à faire ***dire*** aux enfants.
The truth *is easy to make children* ***tell.***
b. ***Les enfants** sont faciles à faire ***dire*** la vérité.
The children *are easy to make* ***tell*** *the truth.*
c. **La glace** est facile à faire ***fondre.***
Ice *is easy to make* ***melt.***
d. ***Les étudiants** sont faciles à faire ***travailler.***
The students *are easy to make* ***work.***

Each diagnostic context can be viewed as a sentence frame with two slots, typographically identified in (2): an **argument slot** for an NP, which is an argument of the verb filling the ***predicate slot***. French (but not English) Object Raising exemplifies the following characteristic property of diagnostic contexts: when a transitive verb is inserted in the predicate slot, the acceptability of the sentence depends on whether the argument slot is filled by the deep subject or the deep direct object of the verb. In (2a), an acceptable sentence results when the deep direct object of ***dire*** "tell" — **la vérité** "truth" — appears in the argument slot, but in (2b) an unacceptable sentence

[marked by "*"] results when the deep subject appears instead — **les enfants** "children". Paralleling this contrast between subject and direct object is a contrast in the behavior of two classes of intransitive verbs. The sole argument of ***fondre*** "melt" is acceptable in the argument slot (2c), while in (2d) the argument of ***travailler*** "work" is not. Since the argument of ***fondre*** is acceptable, like the deep direct object of ***dire***, it is classified as unaccusative; since the argument of ***travailler*** is unacceptable, like the deep subject of ***dire***, it is classified as unergative.

In English, it turns out, diagnostic contexts for unaccusativity are few in number,[2] but this is by no means the case cross-linguistically. The subject of much current research in syntax and semantics, unaccusativity contexts are cross-linguistically a rich source of interesting patterns. Of central concern is the problem of *unaccusativity mismatches*, which are of two general types. First, across languages, synonymous verbs may be unaccusative in some languages and unergative in others. What is relevant to this paper, however, are the language-internal mismatches: the existence of many intransitive verbs that behave unaccusatively in some contexts and unergatively in others.

A good language in which to study language-internal unaccusativity mismatches is French. Legendre (1989, forthcoming) has identified 10 diagnostic contexts in French, and argued that a necessary and sufficient condition for identifying a verb as unaccusative is that it behave unaccusatively in at least one of these contexts. These contexts display a highly complex pattern of mismatches. It is a subset of these data for which we provide a new type of formal account in this paper. For this initial study, we selected four of these 10 diagnostic contexts; these contexts identify the largest numbers of unaccusative verbs. These four contexts are indicated in Table 1; the first is Object Raising, illustrated in (2).

Table 1.

Context	Example	Translation
Object Raising (OR)	**La glace** est facile à faire ***fondre.***	**Ice** *is easy to make* ***melt.***
Croire "believe" (CR)	Je croyais **Marie** déjà ***sortie.***	***I believed*** **Marie** *to have already* ***gone out.***
Participial Absolute (PA)	***Parti*** avant l'aube, **Pierre** est arrivé à destination le jour même.	***Gone*** *before dawn,* **Pierre** *arrived to destination on the same day.*
Reduced Relatives (RR)	**La neige** ***fondue*** a formé de la boue.	*The* ***melted*** **snow** *formed mud.*

In current linguistic theory, the contrasts between unaccusative and unergative intransitive verbs in diagnostic contexts are accounted for by some linguists through syntactic means and by others through semantic means. Syntactic accounts appeal to (1) structural differences between unergatives and unaccusatives, like those posited in (1b,c), (2) structural similarities between the unergative argument and the subject of transitives (1b,a), (3) structural similarities between the unaccusative argument and the direct object of transitives (1c,a), (4) structural restrictions on allowed arguments in diagnostic contexts, and (5) lexical markings indicating which intransitives are unaccusative and which are unergative. Earlier studies led to the claim that the class of unaccusative verbs across languages is not uniform semantically, but uniform syntactically, as expressed in (1c) (Rosen, 1984); however, language-internal mismatches cannot be accounted for on syntactic grounds alone (Legendre, 1989, forthcoming).

Semantic accounts appeal to (1) semantic differences among arguments along various feature dimensions; (2) semantic and aspectual[3] differences among predicates along various feature dimensions; and (3) restrictions in diagnostic contexts on allowed semantic features of arguments and predicates. Semantic features of arguments that are cross-linguistically relevant for unaccusativity include volitionality and animacy, while relevant semantic/aspectual features of predicates include telicity and progressivity,[4] and the classification of verbs as involving activity, accomplishment, achievement, or state (Van Valin, to appear).

Focussing on mismatches, several recent studies (e.g., Zaenen, 1989; Van Valin, to appear) have argued that unaccusativity phenomena can be accounted for on purely semantic grounds by assuming that each diagnostic context involves a simple semantic restriction such as a constraint on the value of some semantic feature. However, this does not appear to be the case for the French data addressed here: at least, the simple semantic restrictions that have been proposed for other languages clearly do not work (Legendre, forthcoming). On the other hand, while they are not sufficient by themselves to provide simple semantic *rules* to account for the data, the argument features volitionality (VO) and animacy (AN), and the predicate features telicity (TE) and progressivity (PR), show strong *tendencies* to influence acceptability in diagnostic contexts (Legendre, forthcoming).

Accounting for the French data appears to require combining syntactic and semantic restrictions in a complex way. This is just what is offered by the connectionist account described here. Replacing the all-or-none categories and rules used in traditional symbolic accounts with formalizations of the semantic and syntactic *tendencies* in the data — *soft rules* — allows our account to achieve a degree of coverage of unaccusativity data, including a complex pattern of mismatches, that, to our knowledge, is unparalleled in the existing literature. At the same time, the new approach naturally addresses two aspects of this problem which are quite difficult to naturally capture in a symbolic approach.

(3) a. ***The graded character of the unaccusative/unergative categories.*** Some intransitive French verbs behave unaccusatively in six of the ten contexts, others four, and still others only one.[5] Clearly the condition of Legendre (1989) — an intransitive verb is unaccusative if and only if it behaves unaccusatively in at least one diagnostic context — is failing to capture the fact that "some verbs are more unaccusative than others," an aspect of the phenomenon that is formally expressed and exploited in our connectionist account.

b. ***The graded character of the acceptability judgements in diagnostic contexts.*** Our account formally predicts not only the polarity of acceptability judgements, but also their strength. While we admit that the accuracy of our account in predicting judgement strengths can stand improvement, it is nonetheless a virtue of our connectionist approach that it makes precise, falsifiable predictions of gradations of acceptability.

Harmonic grammar

The goal of harmonic grammar is to provide a framework, derived from basic connectionist principles, in which regularities in linguistic well-formedness are expressed as *tendencies, preferences* or *soft rules*, rather than as hard rules. This framework is to be a *formal* one in the sense that soft rules are to be specified with sufficient precision to permit precise, falsifiable predictions of the acceptability or well-formedness of sentences.

In its most intuitive form, a central idea of harmonic grammar is to replace hard rules or constraints on well-formedness of the form (4a) with the corresponding soft rule or constraint in (4b).

(4) a. Condition X must never be violated in well-formed structures.

b. If Condition X is violated, then the well-formedness of the structure is diminished by C_X.

For the unaccusativity problem, examples of relevant grammatical and lexical rules are shown in (5) and (6), respectively.

(5) a. In the Object Raising context, the argument can never be a deep subject in a well-formed sentence.

b. In the Object Raising context, if the argument is a deep subject, then the well-formedness of the sentence is diminished by $C_{OR,Subj}$.

(6) a. The argument of *fondre* must not be a deep subject.

b. If the argument of *fondre* is a deep subject, then the well-formedness of the sentence is diminished by $C_{fondre,Subj}$.

Obviously, in harmonic grammar, well-formedness is quantitative; in fact, well-formedness corresponds to the connectionist quantity called *harmony*, and central among the connectionist principles from which the formalism of harmonic grammar is derived are those relating to harmony. The full description of the harmonic grammar formalism, and its derivation from basic connectionist principles, requires combining the description presented here with that found in LMS_2.

The presence of the numerical constants C_X in the soft rules of harmonic grammar would make the formalism hopelessly unwieldy were it not for the fact, discussed below, that these numbers can be computed automatically by presenting the linguistic data to an appropriately designed connectionist network using an appropriate learning rule. (This particular learning procedure is, however, not viable as a model of language acquisition; among other things, it relies on both positive and negative evidence.)

The methodology of harmonic grammar can be described in general terms as follows:

(7) a. For a given problem concerning the well-formedness of certain linguistic structures, choose a rule-based approach (or approaches) to the problem — corresponding to (4a) — and a target set of data that exemplify the problem.

b. Based on the selected rules, identify a set of relevant features for describing the structures appearing in the data.

c. Based on the selected rules, identify which features can plausibly enter together into well-

formedness constraints.

d. Embody these constraints as connections in a network. This network takes as input a featural description of a structure, and produces as output a graded well-formedness or acceptability measure.

e. Train the network on the target data. The resulting connection weights can be interpreted as the constants C_χ in the soft rules of (4b).

f. Analyze the network, extracting from it accounts of the general patterns in the target data that provide new linguistic insights into the original problem.

In the preceding section, we addressed (7a,b) for the specific application of unaccusativity in French. For (7a), we identified rule-based accounts based on structural considerations and on semantic/aspectual considerations, and identified four diagnostic contexts for unaccusativity. For this first research effort, we treat each diagnostic context as a unit, without making use of its internal structure, which is usually quite complex (by connectionist standards). The actual target data is displayed in Table 2. The first four columns correspond to the four contexts, and each row corresponds to argument/predicate pairs with a given set of features that give rise to a certain pattern of acceptability across the contexts. Each entry in the table indicates the acceptability of a given row's argument/predicate pair in a given column's syntactic context. The symbols "+", "+?", "?", "-?", and "-" indicate "acceptable," "marginally acceptable," "of indeterminate acceptability," "marginally unacceptable," and "unacceptable," respectively. The judgements are those of several mutually consistent informants (one of whom is Legendre). Note that these data are what Chomsky (1965) would consider linguistic *performance*, rather than *competence* — *acceptability* rather than *grammaticality* — since they are raw intuitive acceptability judgements from informants rather than outputs from an underlying grammar hypothesized by the linguist and presumed to operate free of obscuring performance factors.

Table 2

OR	CR	PA	RR	Feature	Predicate	OR	CR	PA	RR	Feature	Predicate	OR	CR	PA	RR	Feature	Predicate
-	-	-	-	- - - +	sévir (4)	-?	+	+	+	- - - +	se retirer	+?	+	+	+	+ + +? +?	se recroqueviller
-	-	-	-	- - + -	ouvrir sur	-?	+	+	+	- - + +	s'ouvrir	+?	+	+	+	+ + + ?	sortir
-	-	-	-	- + - -	durer (4)	-?	+	+	+	- + - +	geler	+?	+	+	+	+ + + +	s'exiler
-	-	-	-	- + - -?	baisser	-?	+	+	+	- + + -	parvenir	+	-	-	-	- - - +	souffler (7)
-	-	-	-	- + - +	souffrir (2)	-?	+	+	+	+ + - +	se retirer	+	-	-	-	- + - +	tousser (4)
-	-	-	-	+ - - +	marcher	?	-	-	-	+ + - -	éternuer	+	-	-	-	+ + - +	se taire (3)
-	-	-	-	+ + - -?	déambuler	?	-	-	-	+ + + +	reculer	+	-	-	-	+ + + +	réagir (2)
-	-	-	-	+ + - ?	exagérer	?	+	+	+	- + - +	vieillir	+	+?	+?	+?	- - - +	se modifier
-	-	-	-	+ + - -	être (10)	?	+	+	+	- + + -	naître	+	+?	+?	+?	- - + +	se dissiper
-	-	-	-	+ + - +	agir (29)	?	+	+	+	- + + +	guérir	+	+?	+	+?	- - - +	peler
-	-	-	-	+ + + -	aller	?	+	+	+	+ + - +	monter (2)	+	+?	+	+	- - - +	vieillir
-	-	-	-	+ + + -?	courir	?	+	+	+	+ + + -	entrer (3)	+	+	-	+?	- - + +	pâlir
-	-	-	-	+ + + +	sauter (2)	?	+	+	+	+ + + +	partir	+	+	+?	+?	- - - +	se disperser
-	-	-	+	- - - -	rester	+?	-	-	-	- - - +	durer	+	+	+?	+?	- + + +	changer
-	?	+?	?	+ + - -	tomber	+?	-	-	-	+ + + ?	capituler	+	+	+?	+	- - - +	pousser
-	+	+	+	- + - -	décéder	+?	-	-	-	- + - +?	rougir (2)	+	+	+?	+	- - + -	éclater
-	+	+	+	- + + -	périr	+?	-	-	-	+ + + -	trébucher	+	+	+	+	- - - +?	jaillir (2)
-	+	+	+	+ + - -	s'effacer (2)	+?	+?	+	+?	- - + +	s'éloigner	+	+	+	+	- - - +	tomber (23)
-	+	+	+	+ + - -?	se réfugier	+?	+?	+	+?	+ + - +	s'éloigner	+	+	+	+	- - + -	entrer (2)
-	+	+	+	+ + + -	surgir	+?	+	+?	+	- - - +	surgir	+	+	+	+	- - + ?	sauter
-?	-	-	-	- - - ?	subsister	+?	+	+	+	- - - +	sortir	+	+	+	+	- - + +?	arriver
-?	-	-	-	- - - +	empirer	+?	+	+	+	- - + -	venir	+	+	+	+	- - + +	s'éteindre (12)
-?	-	-	-	+ + - +	sourire (3)	+?	+	+	+	- - + +	s'effacer	+	+	+	+	- + - +	grossir
-?	-	-	-	+ + + +	s'esquiver (3)	+?	+	+	+	- + + -	tomber	+	+	+	+	- + + +	s'évanouir (2)
-?	+?	-	+?	- + - -?	peler	+?	+	+	+	+ + - -	disparaître	+	+	+	+	+ + - +	se réunir (3)
-?	+?	+	+	+ + + +	venir	+?	+	+	+	+ + - ?	se dissimuler	+	+	+	+	+ + + +	se noyer (4)
-?	+	-	+?	? + + +	pâlir	+?	+	+	+	+ + - +	sombrer						
-?	+	-	+	- + - +	grandir	+?	+	+	+	+ + +? ?	se blottir						

For (7b), we identified the structural argument features *deep subject/direct object*, (which we abbreviate "1"/"2"), the argument semantic features *volitional* ("VO") and *animate* ("AN"), and the predicate aspectual features *telic* ("TE") and *progressive* ("PR"). In addition, each individual intransitive verb is individuated for the purpose of enabling its lexical entry to indicate its particular preference for its argument being a 1 or a 2 (as in the syntactic account). In Table 2, at the right of each row is the identity of the predicate preceded by the values of the four semantic/aspectual features (in the order VO, AN, TE, PR). If there are more than one argument/predicate pairs in the category, the identity of one of such predicates is shown followed by the number of pairs in the category in

parentheses. The rows are sorted, essentially going from most unergative to most unaccusative behavior. Table 2 contains 190 rows (argument/predicate pairs) involving 143 different predicates (intransitive verbs). This constitutes acceptability judgements for 760 different sentence types.

In the rest of this paper, we will discuss selected aspects of steps (7c,d,f), and describe step (7e). Most of LMS_2 addresses the crucial steps (7c,d) further, while further development of (7f) is the goal of current research.

The connectionist network

Having set up the linguistic basis for the connectionist account, steps (7a,b), we now describe the connectionist network, illustrated in Figure 1. We begin with a partial description of steps (7c,d), which omits a number of considerations taken up in LMS_2 which depend upon general connectionist principles concerning the distributed representation of structures, the harmony function, and its role in connectionist processing. LMS_2 shows how the network we now discuss is actually mathematically derived as an approximate higher level description of a lower level model based on these connectionist principles, together with the constraints imposed by the elements of linguistic analysis selected in steps (7a-c). This mathematical derivation is responsible for the many aspects of this network which would otherwise seem to be totally ad hoc stipulations.

Input sentences

Each of the 760 input sentences of Table 2 is first coded using a set of variables which are divided into the following four groups:

(8) a. *Context*: the four diagnostic contexts (OR, CR, PA, and RR) are represented by four variables, one for each context. For a given sentence, the variable corresponding to the appropriate context is given the value 1, and the others 0.

b. $A_{feature}$: two variables for the two argument features, VO and AN, each having a value between 1 (+) and 0 (–).

c. $P_{feature}$: two variables for the two predicate features TE and PR, each having a value between 1 (+) and 0 (–).

d. $P_{individual}$: 143 variables each representing one of the 143 predicates. For a given sentence, the variable corresponding to the appropriate predicate is given the value 1, and the others 0.

In the connectionist network, there is a separate input unit for each of these 151 variables (143 identifying individual predicates + 8 others). These units are indicated in Figure 1 by squares. In addition, there is a single output unit whose activity value indicates the network's prediction of the acceptability of the input sentence.

The architecture

The structure of the network is designed to ensure that: (1) the number of free parameters (connection weights) in the network is small compared to the number of data points (760) used to determine them; and (2) each parameter has a clear linguistic interpretation. The following linguistic considerations constrain the architecture of the network:

(9) a. The network should be able to represent the crucial syntactic distinction deep subject ("1") vs. direct object ("2"), which we will call the "structural feature." Since the structural feature is not explicitly present in the input sentence, the network is given two hidden units to use, one representing feature value 1 and the other value 2.

b. For a given context like OR, the grammar can include well-formedness constraints on the structural feature (e.g., "OR prefers 2," as in (6a)) and on the *features* of the predicate (e.g., "OR prefers +PR") *but not* the *identity* of the individual predicate (*not* "OR prefers the predicate *fondre* and prohibits the predicate *éternuer*").

c. For a given context, the grammar can include well-formedness constraints on $A_{feature}$, the argument features (e.g., "OR prefers -VO").

d. A given context can directly bias well-formedness.[6]

e. $A_{feature}$ and $P_{feature}$ can have well-formedness constraints in conjunction with each other (e.g., "+TE prefers +VO") but individually neither one of them can directly constrain well-formedness (e.g., *not* "+AN is absolutely preferred to -AN").

f. $A_{feature}$ and $P_{feature}$ can have preferences for the structural feature (e.g., "+VO prefers 1," "+TE prefers 2").

g. Each individual predicate $P_{individual}$ can have its own preference for the structural feature (a lexical marking corresponding to unaccusative/unergative, e.g. (6b)).

h. Each individual predicate cannot have an absolute bias on grammaticality.

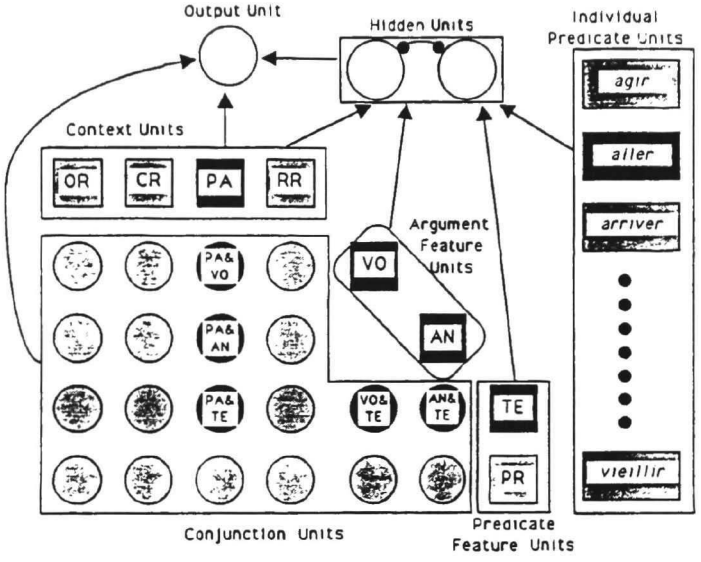


Figure 1. The network architecture.

Together with the connectionist considerations discussed in LMS_2, these linguistic considerations lead to the architecture shown in figure 1. The Conjunction Units represent the constraints that are expressed in terms of conjunctions between pairs of input variables. Each Conjunction Unit is activated in proportion to the product of a particular pair of input units (e.g., "OR's preference for VO" is implemented through a conjunctive unit whose activation is the product of that of the OR unit and that of the VO unit). $P_{individual}$ is not involved in these conjunctions because of the constraint (9b) above.

The Conjunction Units are directly connected to the output unit, following (9b,c,e), as are the Context Units (9d). The Context Units are also connected to the hidden units (9b), as are the Predicate Feature Units, Argument Feature Units, (9f) and Individual Predicate Units (9g). The two hidden units are linear units (their activations are the same as the net inputs). Since the two hidden units are presumed to represent two mutually exclusive structures, they compete against each other, and only the winner of the two (with higher activation) is allowed to send its activation to the output unit. Each connection weight to hidden unit 1 is constrained to be the negative of the corresponding weight to hidden unit 2.[7]

Thus, the net input to the output unit O_{net} is computed as the sum of the contribution of the conjunctions, the contribution of Context Units, and the contribution from the winning hidden unit. As shown in LMS_2, by construction, this net input is equivalent to the global harmony value of the lower level network when it represents the input sentence, which is the network's estimate of the well-formedness of the input sentence. The actual activation value of the output unit, which represents the acceptability estimate, is assumed to be a monotonically increasing function of this net input or global harmony. In the simulation, a logistic sigmoid function $1/(1+e^{-x})$ is used, as is now common in connectionist networks.[8]

The total number of independent connection strengths in the network is 175; 143 embody the lexical preferences of individual verbs for the structural feature, while the remaining 32 are general grammatical preferences.

Training

The network was trained using the informants' acceptability judgements for the 760 sentences shown in Table 2 as the target activation values for its output unit. The standard back propagation algorithm (Rumelhart, Hinton, Williams, 1986) was used to adjust the connection weights which performed a gradient descent (with momentum) on the squared error between the output and the target averaged across all the sentences. For the weights to the hidden units, the winning hidden unit was first allowed to change its in-coming weights, and the weights to the other

hidden unit were then constrained to be the negative of the winner's weights. The direct connections to the output unit were adjusted in the normal way.

Results

As the performance measure, we counted two kinds of errors the network made after training. The network made a *major error* when its output and the target were in the opposite sides of 0.5 (e.g., the output was 0.7 when the target was 0.3). It made a *minor error* when the output differed from the target by more than 0.1. Since the metric of the target values was somewhat arbitrary, we have initially been more concerned with the number of major errors the network made than the number of minor errors. The network tended to be quite close to a solution in less than 1000 iterations through the training sentences and the number of major and minor errors changed little afterwards, but the training was continued until 5000 iterations.

The network's performance varied slightly depending on a few learning parameters. The best network made as few as 2 major errors and 64 minor errors out of the 760 sentences. The worst case was 9 major and 120 minor errors. Despite these variations, the following observations showed consistencies in the solutions. First, the errors were very consistent across solutions. In fact, there was a hierarchy of (major) errors such that if network A made less errors than network B, the errors made by A were always a subset of errors made by B. Second, it is possible to predict what acceptability patterns across contexts are possible for a new predicate/argument pair with given features by varying the connection weight for the predicate. (For example, it might be predicted that only four patterns of acceptability are possible for any predicate/argument pair with the features -VO, +AN, +TE, and +PR out of 2^4 possible patterns.) The networks were fairly consistent in their predicted patterns of acceptability. Third, the network always seemed to solve this task by trying to strike a delicate balance among its weights. For example, it was typical that in order to turn on the output unit by generating +1.5 at the net-input, the network used 8 positive contributions adding up to 32.5 and 7 negative contributions adding up to -31.0. As a result, a decision could be flipped by only a slight change to any of the contributing weights.

Conclusion

We have presented a connectionist network that can be interpreted as a partial grammar and lexicon of French containing 175 soft rules (143 in the lexicon, 32 in the grammar) refering to both structural and semantic features. This network makes precise predictions of the acceptability of a particular range of sentences, and is able to account for the complex pattern of unaccusative mismatches exhibited in 760 sentence types, making only two major errors. The numerical strengths of the soft rules are computed from the data by a connectionist learning algorithm. We are currently in the process of analyzing the network to understand its means of accounting for the regularities in the phenomenon, in the hopes of gaining new linguistic insights. If this analysis proves successful, we believe we will have demonstrated that a new kind of linguistic formalism can provide the current best account of a challenging problem combining syntax and semantics.

This new formalism, harmonic grammar, has a number of advantages. On the one hand, the formalism has several of the attractive features of formal symbolic linguistics; it allows us to:

(10) a. do formal analysis: hypothesize a set of grammatical principles that are sufficient to make definite, falsifiable predictions
b. do explanatory analysis: work within a constrained set of possibilities, and go beyond mere description of the data
c. study constraints on linguistic well-formedness
d. incorporate analytical insights from existing formal linguistics

On the other hand, it has certain characteristics that are often felt to be lacking in formal symbolic linguistics; it allows us to:

(11) a. use data on linguistic performance rather than competence
b. deal with the full complexity of data in a specific area of language
c. work within a formalism that is mathematically derived from connectionism, a cognitively (and statistically) based computational framework; we are in a good position from which to pursue integration with other cognitive language models (e.g., acquisition, real-time processing, and neurolinguistic models)

Acknowledgements

Warm thanks to Alan Prince, for very helpful discussions, and especially, a great pot of chicken soup and the term "isoharmonic." This work owes its existence to Mike Mozer, who failed to convince us not to do it. Thanks also to Jim Martin for his valuable comments on an earlier version. This work has been supported by NSF grants IRI-8609599 and ECE-8617947 to PS, by a grant to PS from the Sloan Foundation's computational neuroscience program. PS (in part) and YM have also been supported by the Optical Connectionist Machine Program of the Center for Optoelectronic Computing Systems, which is sponsored in part by NSF/ERC grant CDR-8622236 and by the Colorado Advanced Technology Institute, an agency of the State of Colorado. GL has been supported in part by a Junior Faculty Development Award from the Council on Research and Creative Work, University of Colorado, Boulder. The authors are listed in alphabetical order.

Footnotes

1. Other connectionist approaches to linguistics appealing to the notion of harmony include John Goldsmith's "harmonic phonology" (Goldsmith, to appear) and George Lakoff's "cognitive phonology" (Lakoff, 1988).

2. One English diagnostic context is the resultative construction (Levin and Rappaport, 1989); under the resultative reading "The door was shut as a result of rolling," the sentence *The door rolled shut* is acceptable, while under the corresponding reading "John was exhausted as a result of working," **John worked exhausted* is unacceptable. This parallels the contrast with transitive verbs between *John wiped the table clean* ("The table was clean as a result of wiping") and **John wiped the table exhausted* ("John was exhausted as a result of wiping"). The argument of intransitive *shut* behaves like the direct object of *wipe* while the argument of *work* behaves like the subject of *wipe*: *shut* is unaccusative while *work* is unergative.

3. Aspect characterizes the internal temporal structure of an event.

4. We take a verb *V* to be telic if "He *V*ed for hours" is unacceptable; we take a verb *V* to be progressive if "He is *V*ing" is acceptable.

5. It is striking that the verbs which are commonly assumed to be prototypical unaccusatives because of their semantic properties actually behave unaccusatively in the fewest numbers of diagnostic contexts. While *exister* "exist" and *être* "be" are strong examples of patient-taking verbs, *exister* behaves unaccusatively in only one context and *être* in only two.

6. These biases are to ill-formedness; a sentence will be unacceptable unless the preferred conditions of the diagnostic context are met, which overcomes the bias. Differences in the strengths of these bias between contexts is one of the factors used to account for mismatches.

7. Because the net inputs to the two hidden units are always the negative of each other, in our implementation we use an equivalent but simpler network in which there is only one hidden unit, which sends the absolute value of its activation to the output unit.

8. The acceptability judgements –, –?, ?, +?, + are taken (rather arbitrarily) to correspond to the numerical values 0.1, 0.3, 0.5, 0.7, 0.9. The same values are used to code the argument and predicate features, except that – is coded as 0 and + is coded as 1.

References

Burzio, L. (1986). *Italian Syntax: A Government-Binding Approach*, Reidel, Dordrecht, Holland.

Chomsky, N. (1965) *Aspects of the theory of syntax*. Cambridge: MIT Press.

Goldsmith, J. (to appear). Harmonic phonology. To appear in J. Goldsmith (ed.) *Proceedings of the Berkeley Workshop on Nonderivational Phonology*. Univ. of Chicago Press.

Lakoff, G. (1988). A suggestion for a linguistics with connectionist foundations. In D. Touretzky, G. Hinton, & T. Sejnowski (eds.), *Proceedings of the 1988 Connectionist Models Summer School*. Morgan Kaufmann Publishers.

Legendre, G. (1989). Unaccusativity in French. *Lingua* **79**, 95–164.

Legendre, G. (forthcoming). *Syntactic vs. semantic accounts of unaccusativity: the need for a new approach.*

Legendre, G., Miyata, Y. & Smolensky, P. (1990) *Harmonic Grammar - A formal multi-level connectionist theory of linguistic well-formedness: Theoretical foundations.* Technical report #90-5, Institute of Cognitive Science, University of Colorado at Boulder.

Levin, B. & Rappaport, M. (1989). An approach to unaccusative mismatches. *Proceedings of the Nineteenth Meeting of the North Eastern Linguistic Society.*

Perlmutter, D.M. (1978). Impersonal passives and the unaccusativity hypothesis. In *Proceedings of the Fourth Berkeley Linguistic Society Meeting.*

Rosen, C. (1984). The interface between semantic roles and initial grammatical relations. In D.M. Perlmutter and C. Rosen (Eds.), *Studies in Relational Grammar 2*, 38–77. Chicago, IL: University of Chicago Press.

Rumelhart, D.E., Hinton, G.E., & Williams, R.J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations.* Cambridge, MA: MIT Press/Bradford Books.

Smolensky, P. (1988). On the proper treatment of connectionism. *The Behavioral and Brain Sciences.* 11, 1–23.

Van Valin, R.D. (to appear). Semantic parameters of split intransitivity. To appear in *Language*.

Zaenen, A. (1989). Unaccusativity in Dutch: An integrated approach. ms, Xerox-PARC and CSLI-Stanford.

PDP Models for Meter Perception

Don L. Scarborough, Benjamin O. Miller,

and

Jacqueline A. Jones

Brooklyn College of the City University of New York

Brooklyn College, Brooklyn, N.Y. 11210

Bitnet: dosbc@cunyvm, bmiller@bklyn, jajbc@cunyvm

Abstract

A basic problem in music perception is how a listener develops a hierarchical representation of the metric structure of music of the sort proposed in the generative theory of Lerdahl & Jackendoff (1983). This paper describes work on a constraint satisfaction approach to the perception of the metric structure of music in which many independent "agents" respond to particular events in the music, and where a representation of the metric structure emerges as a result of distributed local interactions between the agents. This approach has been implemented in two PDP simulation models that instantiate the constraints in different ways. The goal of this work is to develop psychologically and physiologically plausible models of meter perception.

Introduction

Many of the achievements in artificial intelligence have been based on rule-based symbolic approaches. On the other hand, there has been widespread interest in parallel distributed processing (PDP) systems which often use a non-symbolic (or a distributed symbol) constraint- satisfaction approach toward solving problems (Marr, 1982). In an earlier report (Miller, Scarborough & Jones, 1988) we described the **Beats** model which is a rule-based approach that builds on earlier work by Longuet- Higgins & Lee (1982). In this paper we describe a PDP approach to the simulation of musical rhythm perception.

Metric Analysis

Meter, in music, is a perceived pulse that marks off equal temporal intervals in the music. These pulses tend to be grouped perceptually, with the first of each group heard as accented; within groups, pulses appear to alternate between weak and strong in a regular way that reflects a hierarchical organization, as illustrated below. In the notation introduced by Lerdahl & Jackendoff (1983) in their **Generative Theory of Tonal Music** (henceforth, **GTTM**), the numbers at the top represent successive equally spaced points in time, and the dots below are pulses. The first row of dots shows a pulse at each successive point in time and might represent, say, time intervals corresponding to eighth notes. The second row of pulses, corresponding to every other pulse of the first row, groups the pulses in the first row by twos and therefore represents a quarter note pulse rate. The third row represents half notes. Moments with dots at more than one level (e.g. time points 1, 3, 5 and 9) represent perceived stresses relative to others, and in this example, they hierarchically organize the pulses into pairs, pairs of pairs, and so on. Of these levels, the most perceptually salient is what we intuitively call the **beat** or what Lerdahl & Jackendoff call the **tactus**, a level in the hierarchy that corresponds to a moderate foot- tapping tempo. It is important to note that such a hierarchical structure is heard even when listening to a sequence of equally spaced, equally intense tones (Fraisse, 1982).

	Time	1	2	3	4	5	6	7	8	9	10	etc.
	1	.	.	.	.	.	.	.	.	.	.	
Metric Level	2	.		.				.		.		
	3	.				.				.		

A metric structure provides the framework for the emergence of rhythm. For example, syncopated rhythms occur when perceived musical accents are heard as occurring at relatively unstressed times in the metric structure. Thus, musical events are heard within the context of the metric structure. However, a listener must use the same musical events to discover the metric structure in the first place. We are interested in how a listener induces a metric hierarchy while listening to the music.

We have developed a PDP model of the process by which listeners determine the meter of a piece of music. A goal of the model is to generate output that conforms to a well-formed metric hierarchy (Lerdahl & Jackendoff, 1983) under conditions where a human listener would perceive such a hierarchical structure. The simulation model makes a single left-to-right pass through the piece and constructs candidate metric levels.

A Constraint-Satisfaction Approach to Meter Perception

In the BEATS rule-based model of meter perception (Miller et al. 1988), metric levels are generated and selected on the basis of rules that embody expectations about meter, such as well-formedness. BEATS can be said to make decisions. Our PDP model, **BeatNet**, performs more or less the same task not by making decisions but by satisfying a set of constraints. Decision making in rule-based BEATS requires a single agent, or inference engine, with knowledge about states of affairs and knowledge of rules to follow with respect to those states. A distributed constraint-satisfaction approach, by contrast, has no single agent, no centralized knowledge of states, and no rules. Rather, in such a network there are many individual agents or processes; each process acts on the basis of local constraints and knows nothing about the state of other processes but can interact with them. Through this interaction, processes affect one another's behavior in a way that is simple at the level of the processes but complex for the network taken as a whole.

The BeatNet model is based on the idea that every level in a metric hierarchy will correspond to the duration and phase of a single note or small group of adjacent notes somewhere in the piece. This suggests that by determining the durations of notes and note groups we will have a set of candidate metric levels that can be used to construct the correct metric hierarchy. This is in fact the approach we have taken in the BeatNet model. However, many durations in a piece will not correspond to a proper metric level. Therefore BeatNet's task is to identify which durations found in the score belong in the meter.

BeatNet can be thought of as a network of very low frequency oscillators whose periods correspond to the onset-to-onset intervals that occur in a piece. BeatNet makes a single, left-to-right pass though a piece without backtracking. It ignores all information other than onset-onset intervals. As BeatNet moves through the score, the time between one note onset and another leads to the excitation of an oscillator, or metronome, a process that embodies this periodicity. A metronome is defined by where it starts, i.e. its phase, and the aggregate duration of the note(s) it spans, i.e. its

period. As a piece is heard, a subset of the oscillators is excited by the time intervals that occur. The activation of a metronome oscillator, then, depends on the detection of a specific time interval or period in the stimulus. If a particular period is detected with more than one phase, each period/phase combination activates a corresponding oscillator with the corresponding period and phase. Oscillators not set in motion are generally inactive.

Simply activating the metronome oscillators that correspond to onset-onset intervals does not lead to a coherent metrical structure. For example, a series of eighth notes will activate metronomes corresponding to an eighth note, a quarter note, a dotted-quarter note, a half note, and so on at various phases. However a metric analysis that conforms to the Lerdahl & Jackendoff rules will only allow a subset of the metronomes that form a well structured hierarchy. This requires that metronomes that appear to be inconsistent with the metric structure be inhibited, while metronomes that are consistent should be further strengthened. To do this, we let metronomes interact via excitation and inhibition in ways that represent constraints on the network that are likely to lead to a coherent metric structure.

In BeatNet, each metronome has a variable strength, or activation level, which is modified in four ways. A metronome's strength decays steadily over time, and is also lowered from time to time by inhibitory inputs from other metronomes. Activation is increased by excitatory inputs from other metronomes. Finally, a metronome's activation increases if it predicts a note onset, i.e. it ticks concurrently with a note onset. Strength is indirectly affected by two other features of BeatNet. First, each metronome has a threshold. If its strength falls below threshold, a metronome can neither inhibit nor excite another metronome, but it continues to tick and can still receive excitation and inhibition. Second, the notion of the tactus, described earlier, suggests that metronomes with moderate periods should have more influence in the network than those with very large or small periods. Accordingly, several parameters are scaled so as to increase the influence of metronomes close to the tactus.

We have created two versions of this BeatNet metronome model. In the first model, called the **Broadcast** model, we have tried to build in very few of the required constraints into the network structure. That is, we configured the network based on a few simple principles that only indirectly embody the constraints that a well formed hierarchy must have. The issue was whether such a simple architecture might nonetheless produce metric analyses that were well formed in the Lerdahl & Jackendoff sense. The second model, called the **Resonance** model, makes much stronger assumptions about the nature of the interactions between metronomes.

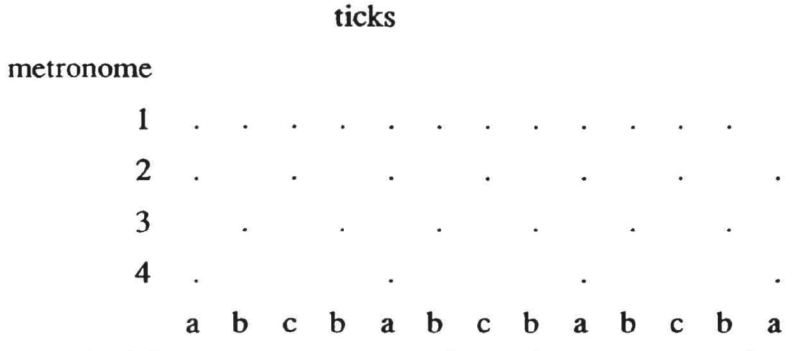
The Broadcast Model

In the Broadcast model, we tried to avoid assuming specific patterns of hardwired interaction among the metronomes. Instead, interaction among metronomes occurs only in response to signals that are broadcast throughout the system in response to particular events. These broadcast signals originate from both external and internal events. An externally generated signal is broadcast whenever a note onset occurs. The effect of this signal is that all metronomes that time out or tick at that instant are activated. We also assume that metronomes differ in sensitivity. That is, we assume that, as with pitch perception, we are most responsive or sensitive to events of a particular period, and that this peak sensitivity is the basis for the tactus, the metric level that is perceived as most salient, and which is generally around 2 Hz (Fraisse, 1947-48). The second source of broadcast

signals comes from the internal ticking of the metronomes themselves. A tick of a metronome is, functionally, much like a note onset in that it excites other metronomes that tick at the same time. The strength of this signal depends on how many metronomes tick at the same time. However, in the case of these metronome ticks, we assume that the metronomes are organized into a linear structure like a basilar membrane, and that the effect of one metronome on another decreases with the distance between the two. Close neighbors are excited or inhibited strongly, while more distant metronomes are hardly affected. Distance, for purposes of calculating the effect of one metronome on another, is defined in terms of the ratio of the periods of the two metronomes.

The idea that metronomes that tick at the same time strengthen each other is a weak way to implement the constraint that in a well formed hierarchy, a metronome with a period T should be in phase with other metronomes whose periods are related to T by simple integer ratios, e.g. 2T or 3T, and 1/2T or 1/3T. Metronomes that are in phase with each other are more likely to tick together than out-of-phase metronomes. In addition, the tick of a metronome broadcasts an inhibitory signal to all metronomes with shorter periods that do not tick at that time. This implements the constraint that in a well formed metric hierarchy, a pulse at one metric level should coincide with pulses at all the lower metric levels. The asymmetry between excitation and inhibition can be seen in the nature of a well-formed metric hierarchy: a beat at a given level need not be a beat at higher levels, but it must be a beat at all lower levels (Lerdahl &Jackendoff, 1983). The Broadcast model's pattern of inhibition and excitation is such that metronomes that together constitute a well-formed hierarchy will tend to strengthen one another and inhibit outsiders. Finally, as mentioned above, we assume that metronomes tend to run down. That is, without sustaining input events, the activation of a metronome will decay.

The interactions can be summarized as follows:



At the 'a' steps metronomes 1, 2 and 4 excite one another and metronome 3 is inhibited by 4, which has a larger period, and by 2, which has the same period and a different phase. At the 'b' steps metronomes 1 and 3 excite one another while 3 inhibits 2, and at the 'c' steps there is only excitation between metronomes 1 and 2. Nothing inhibits metronome 4, because it is the largest, or metronome 1, because it ticks together with all higher-level metronomes.

Figure 1 illustrates metronome interactions. Each curve represents a single metronome. The ordinate represents the activation level of a metronome. The abcissa intersects the ordinate at an activation level of 0. The horizontal dashed line above the abcissa represents the threshold that a metronome must reach before it can affect other metronomes. The abcissa represents time, and the vertical dashed lines mark note onsets. The very first note onset occurs at the ordinate. Figure 1a (left panel) illustrates the emerging activation pattern that occurs in various metronomes when the

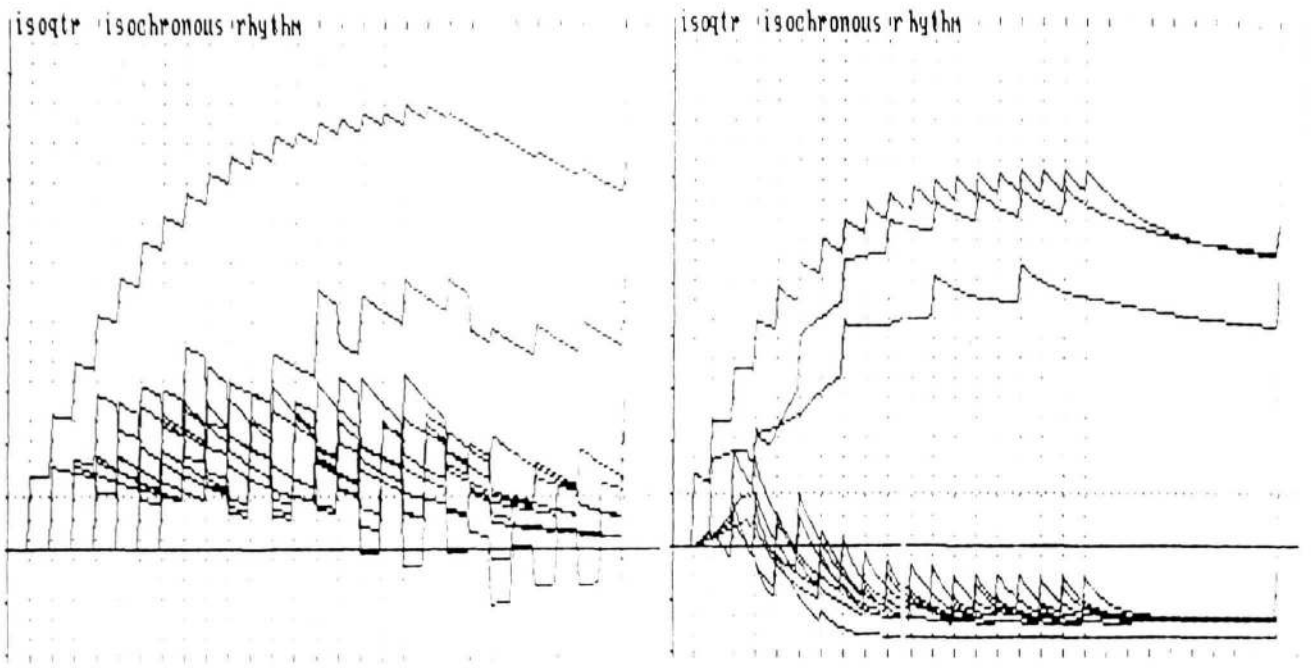


Fig. 1, Quarter Notes: a) Broadcast; b) Resonance

Broadcast network hears an isochronous sequence of 20 quarter notes followed by two whole rests. The jaggedness of the activation patterns occurs because of changes in activation that occur in response to external and internal signals. In between these signals, activations decrease as a result of decay. Though this pattern appears to be rather chaotic, a quarter note metronome (top curve) and a half note metronome (second highest curve) clearly emerge from the noise. Further, these two metronomes are in phase with each other. Less clearly evident is a whole note metronome. It too is in phase with the two more active metronomes. However, it is in competition with other whole note metronomes that differ in phase. In addition, there are metronomes with a period of three quarter notes at various phases. No clear winner emerges from this competition. However, this is not necessarily a failing of the model for two reasons. First, with an isochronous sequence, there is no clear basis for inducing one well- formed metric structure over another. Second, the lack of clarity at higher levels of the metric structure matches the idea that the tactus is an intermediate level of the metric structure that is most strongly perceived.



Fig. 2, London Bridge is Falling Down

Figure 2 shows the note duration sequence for the song, London Bridge is Falling Down, which is in 4/4 time, with the first dotted quarter note beginning on the down beat. The response of the Broadcast model to this piece is shown in Figure 3a (left panel). The top line represents a quarter note metronome, and the next two lines down are an eighth note and a half note metronome, respectively. These three metronomes form a proper metric hierarchy. No whole note metronome emerges clearly, but the next strongest metronome is, in fact, a whole note metronome with a phase that agrees with the other levels of the metric structure.

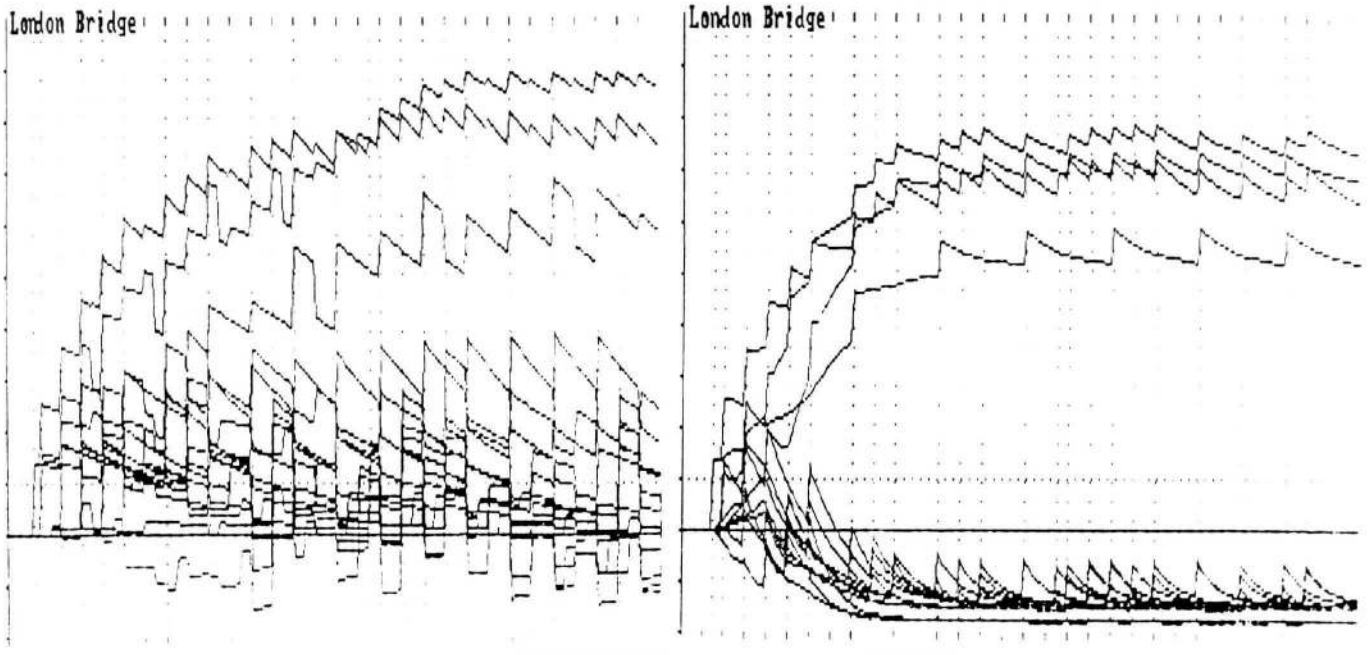


Fig. 3, London Bridge: a) Broadcast; b) Resonance

All in all, the Broadcast model seems to do a fairly good job of inducing appropriate metric structures on the basis of some fairly simple assumptions about how the metronomes might interact. Further, the interaction pattern does not appear to depend critically on the choice of parameters that determine the magnitude of the interaction effects within fairly wide limits. The important point is that the principles for metronome interaction do not explicitly embody the concept of a metric hierarchy.

The Resonance Model

In the Resonance model, as with the Broadcast model, we assume that a metronome is excited by the occurrence of note-onset intervals that match its period and phase. However, the Resonance model makes more specific assumptions about the architecture. In particular, this model assumes that excitatory and inhibitory interactions continue to occur in the interval between ticks and note onsets. For this to occur, we have to assume that each pair of metronomes is connected via an

excitatory or an inhibitory connection depending on whether or not that metronome pair is in phase, and on whether their periods form a simple integer ratio. This assumption leads to a pattern of interactions that produces much quicker sorting out of "bad" metronomes. In addition, depending on the strength of the interactions, the Resonance model literally does resonate in that a particular subset of the metronomes can form a pattern of interactions in which the strength of the internal signals alone is sufficient to hold the metronomes in a stable pattern of activation. This may not be an unreasonable characteristic of the model because it means that the same structure that recognizes a metric structure can also produce a metric structure, just as a listener, after hearing a few bars of a piece, can anticipate the rhythmic pattern thereafter.

Figure 1b (right panel) illustrates the emerging metric structure with the Resonance Model in response to the isochronous quarter note sequence. Here, an unambiguous metric structure emerges consisting of a quarter note, a half note, and a whole note metronome all with the appropriate phase relationship. All other candidate metric levels become quickly inhibited. Figure 3b (right panel) shows the response of the Resonance model to London Bridge. Here too a complete metric hierarchy emerges. The top four lines represent a quarter note, an eighth note, a half note and a whole note level. Again, other candidate metronome levels are strongly inhibited within a relatively few notes.

The Resonance model induces metric structures much more clearly than does the Broadcast model. However, before we can entertain the Resonance model as a plausible description of how people might recognize metric structures, we need to have an account of how the specific pattern of excitatory and inhibitory weights between metronomes that is involved in this model might arise. At this point, we do not have a clear answer to this issue.

Conclusion

The BeatNet models succeed in producing reasonable metric analyses for many pieces. In other cases, the output does not coincide with the interpretation specified by the score. However, it is possible that a human listener would make the same errors if deprived of all information other than time intervals. While BeatNet has some intuitively appealing properties, it is not without problems. A difficulty with BeatNet that is inherent in network models is the virtual impossibility of predicting the model's output and the difficulty in understanding the relation of this output to the system parameters. However, in general we have tried to constrain the system parameters based on a priori judgments about what was reasonable. On the other hand, while a rule-based system such as BEATS (Miller et al. 1988) has few parameters and is very predictable -- on the whole it can produce only one analysis -- BeatNet can produce different output as the values of its initial state and parameters are changed. Exploring the various configurations over a number of parameters is a daunting task, but we are encouraged by the fact that a configuration that works well with one score tends to work well with other scores.

Acknowledgments

This work was supported by an NSF Pre-Doctoral Fellowship to Miller and by PSC-CUNY Faculty Research Awards to Jones and Scarborough.

References

Fraisse, P. (1947-48). Mouvements rythmiques et arythmiques. **L'Annee Psychologique, 47-48**, 11-21.

Fraisse, P. (1982). Rhythm and tempo. In D. Deutsch (Ed.), **The Psychology of Music** (pp. 149-180). New York: Academic Press.

Lerdahl, F. & Jackendoff, R. (1983). **A Generative Theory of Tonal Music**. Cambridge: MIT Press, 1983.

Longuet-Higgins, H.C. & Lee, C.S. (1982). The perception of musical rhythms. **Perception, 11**, 115-128.

Marr, D. (1982). **Vision**. San Francisco: W.H. Freeman.

Miller, B., Scarborough, D. & Jones, J. (1988). A model of meter perception in music. **Proceedings of the Tenth Annual Conference of the Cognitive Science Society**. (pp. 717-723). Hillsdale, NJ.: Lawrence Erlbaum Associates.

Some Criteria for Evaluating Designs*

Thomas R. Hinrichs
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, Georgia 30332

Abstract

Most non-trivial design tasks are *under-specified*, which makes evaluating designs subjective and problematic. In this paper, we address the evaluation criteria that are left implicit in problem specifications. We propose that these criteria evaluate designs in terms of specific types of consistency and completeness. In particular, we divide consistency into constraint, representational, and goal consistency, and we decompose completeness into the specificity, depth, and breadth of a solution. These distinctions are useful because they organize criteria for evaluating designs. This model of evaluation is largely implemented in a program called JULIA that plans the presentation and menu of meals to satisfy multiple, interacting constraints.

1 Introduction

Real world design problems are almost universally under-specified. Consequently, designers must augment formal specifications with their own expectations and criteria for evaluating designs. What are these criteria and where do they come from? One possibility is that they are simple syntactic checks on the consistency and completeness of a design. The problem with this approach is that it is insufficiently flexible, since the appropriate level of completeness may vary for the same problem in different contexts. For example, generating a shopping list for a meal requires a more specific menu than deciding whether that same meal will appeal to guests. In short, to evaluate a design, one must know the requirements of the design itself, as well as those of the artifact.

Previous research in design has emphasized the search strategies that designers use rather than their evaluation criteria and stopping rules. This may be because the tool of

*This research was funded in part by NSF Grant No. IST-8608362, in part by DARPA Grant No. F49620-88-C-0058

choice in studying design is *protocol analysis*, which facilitates analyzing the time-course of problem solving, but provides little help in eliciting underlying decision criteria. Protocol studies have been reported in a variety of design domains, including architecture [Eastman 1969, Goel & Pirolli 1989], mechanical design [Ullman *et al.* 1986], software design [Malhotra *et al.*, 1980], algorithm design [Kant & Newell 1982], and meal planning [Byrne 1977]. In each of these studies, problem specifications were informal and incomplete, yet designers were able to evaluate their solutions. The unarticulated evaluation criteria which they used are the subject of this research.

In this paper, we define different types of consistency and completeness. These distinctions are useful because they serve to organize criteria for evaluating designs. Criteria for *consistency* enable the evaluation of partial designs and the coordination of multiple representations. Criteria for *completeness* provide contextual clues for determining when to stop refining a design, thereby permitting the design process to be iterative.

In the next section, we discuss criteria for evaluating the consistency of designs. In section 3, we elaborate the notion of completeness and present criteria for evaluating the specificity and scope of designs. These evaluation are being implemented in a computer program called JULIA [Hinrichs 1988, Hinrichs 1989], which plans meals to satisfy multiple, interacting constraints.

2 Evaluating Consistency

Consistency maintenance is the means by which a designer ensures that different parts of a solution agree with each other. Two factors conspire to make consistency maintenance difficult. First is the need to evaluate *partial solutions*. When a design is only partly refined, missing information may make constraints appear violated when they are not. For example, a global constraint that a meal contain eggplant may appear to be violated until a main dish is chosen. Alternatively, missing information may camouflage an actual inconsistency. For example, caviar may not appear inconsistent with a tight budget if the rest of the menu is not specified yet. To address this issue, we present an approach to constraint management that permits constraints to be incrementally formulated, evaluated under different policies, and strategically relaxed.

The second factor is called the *Logical Omniscience Fallacy* [Halpern 1986]. This is the mistaken assumption that it is reasonable for an agent to know all the implications of his beliefs. This assumption often shows up in Artificial Intelligence models of problem solving in which consistency is maintained via a single, uniform mechanism such as a Reason Maintenance System [Doyle 1979]. The problem with this is that it is both overly powerful and insufficiently flexible. In particular, it assumes that there is only a single type of consistency. However, design involves several different types of relationships that have independent and distinct criteria for evaluating consistency. To accommodate this, we partition design relationships into 1) *constrained* relationships between parts of a solution,

2) relationships between *representations*, and 3) relationships between *goals*. Consistency across each type of relationship is maintained by independent, dedicated mechanisms.

2.1 Constraint consistency

Most of the overt criteria for evaluating a design involve the satisfaction of constraints. These criteria determine *which* constraints are to be satisfied, to *what degree* they are satisfied, and what must be *changed* to maintain consistency. Constraint consistency is implemented in JULIA by a constraint poster that is built on top of a Reason Maintenance System.

Where do constraints come from?

To evaluate solutions to under-specified problems, a designer must implicitly fill in constraints that are missing. The process that generates these constraints is called *formulation* [Stefik 1981]. Constraint formulation is an incremental process in which the problem specification is refined as commitments are made in the solution. The constraint formulation process generates constraints from three sources:

- *Problem statement.* Many constraints are explicit in the original problem specification. For example, a typical problem that JULIA might solve is: "Plan a vegetarian Greek birthday dinner that contains eggplant."
- *General plans.* Some constraints are imported from general schematic knowledge of partial solutions. For instance, individual meals in JULIA inherit the constraint that side dishes should be compatible with the main dish.
- *Other constraints.* Some constraints are inferred from other constraints. For example, JULIA knows about the food preferences of individual people. These preference constraints are propagated to the meal being planned.

Strategies for evaluating constraints

A designer must be able to evaluate partial solutions, consequently it should be possible to check constraints when some information is missing. To do this, our model of design provides explicit strategies for evaluating constraints when some information is missing, or when approximate solutions are acceptable. These strategies include:

- *Optimistic evaluation.* Design constraints can be evaluated optimistically by accepting solutions for which constraints are not explicitly violated. JULIA adopts this policy when generating potential solutions.

- *Pessimistic evaluation.* Alternatively, potential solutions can be rejected whenever information with which to evaluate a constraint is missing. JULIA uses pessimistic evaluation when comparing two potential solutions, in order to determine which alternative satisfies constraints 'better'.
- *Approximate evaluation.* When constraints can be partially satisfied, a simple predicate is insufficient for evaluation. Instead, evaluation must indicate the degree to which constraints are satisfied. JULIA considers partial satisfaction when relaxing constraints on an over-constrained problem.

Strategies for resolving inconsistency

When constraints contradict each other, the inconsistency must be resolved. The ways in which inconsistencies are resolved can serve as evaluation criteria for the overall quality of a solution. For example, the fewer the constraints relaxed, the better the design. Three general strategies for resolving inconsistencies are:

- *Change the solution.* Inconsistent solutions can be corrected by either retracting a previous decision or by applying some new transformation to adapt it.
- *Change the problem.* Sometimes, it is not possible to solve a given problem. In this case, one strategy is to simplify the problem until it can be solved. This can be done by relaxing preference constraints.
- *Ignore it.* Some inconsistencies can be recognized and yet ignored, such as preference constraints that cannot be satisfied.

2.2 Representational consistency

The second type of consistency we distinguish is consistency across different *representations* of a design. Any problem solver that uses multiple representations must ensure that the representations agree with each other. JULIA maintains two types of representation, an *assertional representation* in the form of reason-maintenance nodes, and a *structural representation* in the form of frames. Representational consistency means that information represented in one form corresponds with, but does not contradict, information represented in another.

If the correspondence between representations becomes ambiguous, some mapping must be chosen. For example, if JULIA is interacting with a client to plan a meal, the client may suggest a particular dish. This suggestion is represented as an assertion that is ambiguous with respect to its role in the structure of the meal (*e.g.* Should it be an appetizer or a side dish?) To maintain representational consistency, JULIA must make an educated guess about the role that the dish is to play in the meal.

If representations conflict, it is necessary to either retract an assertion, or relax a representational assumption. For example, when JULIA asks a client to choose between two dishes, the client may answer 'both'. In this case, the structural representation assumes that there will be a unique dish in the role. JULIA must therefore change the structural representation to accommodate multiple dishes. Representational consistency is maintained by a module called a *structure maintenance system*, or SMS.

2.3 Goal consistency

The third type of consistency we distinguish is consistency across the *goals* of the designer. Unlike constraint and representational consistency, conflicting goals are an inescapable part of design. Designers must be able to entertain conflicting goals, because designs are often specified in contradictory terms. For example, the specification for an automobile may require that it be inexpensive, yet safe.

Nevertheless, some goal relationships *are* maintained. First, the designer's goals should track the structure of the evolving solution. To do this, when new structures are added, the problem solver posts new goals to refine the structure, and when structures are deleted, it abandons the corresponding goals. Second, when a goal is abandoned, its subgoals are also abandoned. In JULIA, these inferences are the responsibility of a *goal scheduler* that maintains an agenda and a network of goals.

3 Evaluating Completeness

The second main criterion for evaluating designs is *completeness*. How complete must a design be in order to be acceptable? Previous models of design have referred to *stopping rules* that terminate design refinement [Byrne 1977, Goel & Pirolli 1989], but they have been vague about just what these rules might be. In order to clarify the ways in which stopping rules may differ, we describe the completeness of designs in terms of their *specificity* and *scope*.

3.1 Criteria for specificity

The *specificity* of a design indicates the level of abstraction of the components of the design. For example, sometimes it is sufficient to indicate a hot soup, while at other times, it is necessary to specify 'Campbells tomato soup.' Criteria for determining the appropriate level of specificity include:

- *Convention.* Solutions should be of approximately the same level of specificity as previous similar designs. This criterion can be measured by reference to some level in

a taxonomy, such as a top level, a terminal level, or a designated basic level.

- *Determining constraint satisfaction.* Solutions should be specific enough that their constraints are explicitly satisfied. This criterion can be measured by the information content of the representation. For example, a requirement on the specificity of a dish might say that it must specify its ingredients. Under this rule, 'soup' would be inadequate, but 'tomato-soup' would suffice.
- *Cognitive load.* Solutions should be general enough to minimize the number of alternatives to be maintained in working memory, or to be communicated to a client. This can be measured by a bound on the size of the contrast set. For example, a meal might be described at the level of 'clear soup' *vs* 'cream soup' simply because the number of alternatives is small.

3.2 Scope

In addition to the specificity of solution components, a design can be evaluated in terms of the adequacy of its structure, or its *scope*. The scope of a design defines the set of variables the designer must solve. Some problems, such as *parametric design* problems, have fixed scope. Many design problems, however, do not have a fixed solution structure and require the designer to evaluate the scope of proposed solutions. The scope of a design can be further decomposed into its *depth* and *breadth*, which more precisely characterize its structure.

Some evidence for the distinction between specificity and scope can be observed in published design protocols. For instance, in [Byrne 1977] and [Kant & Newell 1982], the experimenters had to occasionally prompt their subjects to be more specific. However, there is no evidence of subjects being confused about the *scope* of their task. This suggests that the evaluation criteria for design scope and specificity are different and may be a function of different types of knowledge.

Criteria for depth

The *depth* of a design describes how finely it is decomposed. For a meal-planner, a shallow design might designate what courses are in a meal, or what dishes will be served in a course. A very deep design might decompose into discrete actions, such as going up to a buffet table, serving, eating, and so forth. Some criteria for determining the appropriate level of granularity of a design or plan are:

- *Simulation.* If there are constraints on the behavior of an artifact, then evaluation may involve simulating its operation. Simulation may require the design to be decomposed to a certain uniform depth.

- *Explanation.* In order to answer questions about a design, it may be necessary to decompose it to finer granularity. For example, to explain why a previous design failed, it may be necessary to simulate it.
- *Convention.* In the absence of other criteria, decompose a solution to some standardized level, based on similar designs or on generic design plans.

Criteria for breadth

The *breadth* of a design characterizes the range of considerations subsumed under the design problem. For example, the breadth of a design for an artifact may range from a simple structural description, all the way to an account of how it will be manufactured, the tools required to build it, and the procedures needed to operate and repair it. These peripheral aspects of a design provide constraints on the more central structure of the design. Criteria for determining the required breadth of a design include:

- *Standardized checklist.* Design tasks that are routine or well-understood often have explicit criteria for completeness in the form of standardized checklists. These serve as external aids to memory.
- *Avoiding failure.* One criterion for the breadth of a design is that it be broad enough to anticipate possible failure modes. To determine this, feedback from previous failed designs can be used to indicate peripheral features that led to goal failures. For example, a design that was impossible to maintain or test should clue the designer to consider these features in the future.
- *Determined by oracle.* An external agent, such as the client, may ask specific questions about peripheral aspects of a design, which would drive the designer to refine these features.

4 Discussion

We have presented a set of criteria for assessing the consistency and completeness of designs. An implication of these criteria is that to evaluate a design, one must know the requirements for the design itself, as well as those of the artifact. One way this idea can be operationalized is to associate evaluation criteria with explicit design goals, so that the requirements for the design can be inferred from the goal network. JULIA has a vocabulary of design goals that would facilitate this, though the completeness criteria have not yet been implemented.

This paper describes work in progress. JULIA exists and designs meals, and the partitioned approach to consistency maintenance is implemented and works. The criteria

for evaluating completeness are only partly integrated into JULIA and are our current focus of research. Some of the problems that remain to be addressed in this work are:

- *Partially satisfiable goals* complicate stopping rules. For example, if a goal of a meal is to minimize calories, then when is a meal sufficiently low-calorie? A design may be adequately consistent and complete, yet still be insufficient.
- *Representing completeness criteria.* Should specificity be represented as a level in a taxonomy, or as a constraint on information content? How should the required depth of a solution be represented? Design evaluation would appear to require a *meta-level* representation.

References

R. Byrne. Planning meals: Problem solving on a real data-base. *Cognition*, 5:287–332, 1977.

J. Doyle. A truth maintenance system. *Artificial Intelligence*, 12(3), 1979.

C.M. Eastman. Cognitive processes and ill-defined problems: A case study from design. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 669–690, 1969.

V. Goel and P. Pirolli. Motivating the notion of generic design within information-processing theory: The design problem space. *AI Magazine*, 10(1), Spring 1989.

J.Y. Halpern. Reasoning about knowledge: An overview. In *Theoretical Aspects of Reasoning About Knowledge*, pages 1–17, 1986.

T.R. Hinrichs. Towards an architecture for open world problem solving. In J.L. Kolodner, editor, *Proceedings of the 1988 DARPA Workshop on Case-Based Reasoning*, pages 182–189, 1988.

T.R. Hinrichs. Strategies for adaptation and recovery in a design problem solver. In K. Hammond, editor, *Proceedings of the 1989 DARPA Workshop on Case-Based Reasoning*, pages 115–118, 1989.

E. Kant and A. Newell. Problem solving techniques for the design of algorithms. Technical Report CMU-CS-82-145, Carnegie Mellon University, 1982.

A. Malhotra, J.C. Thomas, J.M. Carroll, and L.A. Miller. Cognitive processes in design. *International Journal of Man-Machine Studies*, 12(2):119–140, 1980.

M.J. Stefik. Planning with constraints. *Artificial Intelligence*, 16(2):111–140, 1981.

D.G. Ullman, L.A. Stauffer, and T.G. Detterich. Preliminary results of an experimental study of the mechanical design process. Technical Report 86-30-9, Computer Science Department, Oregon State University, 1986.

Ad-Hoc, Fail-Safe Plan Learning

Roland Zito-Wolf
Richard Alterman

Computer Science Department
Brandeis University
Waltham MA 02254

Abstract

Artificial Intelligence research has traditionally treated planning, execution and learning as independent, sequential subproblems decomposing the larger problem of intelligent action. Recently, several lines of research have challenged the separation of planning and acting. This paper suggests that integration with planning and acting is also important for learning. We present an integrated system SCAVENGER combining an adaptive planner with an ad-hoc learner. Situated plans are retrieved from memory; adaptation during execution extends these plans to cope with contingencies that arise and to tease out descriptions of situations to which these plans pertain. These changes are then integrated into the plan and incorporated into memory. Every situation of action is an opportunity for learning. Adaptive planning makes learning *fail-safe* by compensating for imperfections and omissions in learning and variability across situations. We discuss a learning example in the domain of mechanical devices.

1 Introduction

Traditional models of planning in artificial intelligence have two key features. First, planning is *ahistoric*: the planner confronts each task in isolation. Plans are constructed from scratch using a limited repertoire of operators and a domain model defining their effects and interactions. Second, planning is *detached from acting*. The plan is constructed first, then given to an independent execution monitor that performs the specified operations. These assumptions help decompose the planning task, but unfortunately introduce other problems. Experience is ignored, leading to redundant planning. Separate execution makes it difficult to adequately address run-time contingencies and uncertainty. This critique of traditional planning models arose from work on adaptive planning[Alterman, 1988], case-based planning[Kolodner and Simpson, 1989; Hammond, 1986], reactive planning[Firby, 1987; Georgeff and Lansky, 1987], and situated activity[Suchman, 1987; Agre and Chapman, 1987; Agre, 1988].

Adaptive planning [Alterman, *ibid.*] was an early effort to address these problems. An adaptive planner is a commonsense planner. It retrieves from memory a plan that fits the situation, and adapts it during engagement – while focused on the task and immersed in the details of the situation – in conjunction with its developing interpretation of the situation.

Adaptive planning provides a congenial environment for learning. Adaptation is a rich source of ideas for plan modification; every situation of engagement is an opportunity to learn. Learning, planning, and action are integrated: planning and acting in particular situations generate data to be learned, while learning is distributed across many situations. Learning extends plans to cope with new contingencies that may arise and teases out descriptions of situations to which a given plan pertains.

This paper presents a system SCAVENGER combining an adaptive planner with a learner. SCAVENGER continually refines its knowledge with experience. Faced with a domain for which the system possesses no complete theory, what else can it do? Learning is *ad-hoc* because what is learned depends on both the particular situation and the learner's current knowledge; knowledge is forged by experience, grounded rather than abstract. We use the term *fail-safe* to suggest not infallibility but rather that adaptation functions during execution, compensating for imperfections and omissions in learning, and insulating the learner from small variations across otherwise similar situations. Learning is the accumulation of detailed experience that can

be adapted to future situations.

SCAVENGER explicitly frames the learning problem to tolerate inconsistency and revision. This is an atypical assumption for planning and learning systems. Planners in the tradition descending from STRIPS[Fikes and Nilsson, 1971; Chapman, 1987] require a strong domain theory to predict the consequences of actions. Recent EBL techniques[Mitchell *et al.*, 1986; DeJong and Mooney, 1986; Mooney, 1988; Minton *et al.*, 1989] require a strong domain theory to guide explanation and constrain generalization. Pazzani[1988] shows that a domain theory for EBL can be constructed inductively; however, the implications of basing EBL on a changing or inconsistent theory are unclear. Our work assumes that knowledge of the domain is inevitably incomplete.

SCAVENGER differs from other memory-based planners that learn, such as MEDIATOR[Kolodner and Simpson, 1989] and CHEF[Hammond, 1986], largely because plan adaptation occurs throughout the period of engagement. For example, contrast the frameworks of CHEF and SCAVENGER. Unlike CHEF, SCAVENGER does not require a complete domain theory or abstract repair schemata. Where CHEF learns about plans, SCAVENGER learns about plans and their situations-of-use. SCAVENGER's adaptations are motivated by the resources available in the situation, rather than derived primarily from operationalization of the domain theory. SCAVENGER is able to acquire many independent items of knowledge in the course of a single trial.

2 SCAVENGER

The FLOABN project[Alterman and Zito-Wolf, forthcoming] addresses planning and plan acquisition in the the domain of everyday mechanical devices such as telephones, clocks and VCRs. This domain is challenging because there is much variability among devices and because only a minimal domain theory is available. In compensation, the execution environment is benign (errors in operation are normally recoverable) and there are multiple opportunities for interaction with each device. SCAVENGER, the core of FLOABN, is an adaptive planner coupled with an ad-hoc learner. Other modules of FLOABN are concerned with skimming instructions for adaptation ideas and simulating in detail the perceptual and effective interactions between agent and device. We are prototyping SCAVENGER in Quintus MacProlog.

A problem is posed to SCAVENGER by specifying a goal and a situation. SCAVENGER's semantic memory defines concepts; its episodic memory contains plans and expectations (see Figure 1). A simulator models the outcome of SCAVENGER's actions in the world. SCAVENGER has no access to the internals of this model; the outcomes of actions in the simulated world may differ from the system's expectations.

Adaptation is driven by *situation differences*: discrepancies between what is expected at each plan step and what is actually observed. Situations may differ in goals, preconditions, and outcomes; further detail can be found in Alterman[1988]. When such a difference is detected, SCAVENGER searches for an adaptation that will either account for or repair it. Adaptations are suggested using general and domain-specific knowledge and related known plans. Adaptation suggestions can also come from other modules of FLOABN; we will not discuss those here. Plans are modified via some combination of inserting, deleting, reordering, substituting and modifying steps. SCAVENGER can also satisfy missing preconditions and outcomes by substitution of similar features or outcomes. This is in many cases more natural than modifying an entire step; more importantly, this allows SCAVENGER to create new steps by modifying known ones. SCAVENGER keeps track of the adaptations made; afterwards, successful adaptations are used to *elaborate* the plan. Elaboration conditionalizes the plan so that it will act appropriately in situations resembling the newly learned situation. Execution of the plan in such circumstances will then entail little or no deliberation.

We demonstrate this process with an example of SCAVENGER learning to use a (simulated) touch-tone phone. SCAVENGER adapts its dial-phone procedure using a variety of clues: the common features shared by touch-tone and dial phones, knowledge of which features are pertinent in a situation, knowledge of which features participate causally in the plan, knowledge about the function of individual steps *within* the plan, and background knowledge about the world. SCAVENGER's learning module then incorporates the successful adaptations into the existing plan and concept structures. This results in the creation of a new device category for touch-tone phones, a generalized category subsuming both phone types, and plan steps specific to touch-tone phones. Lastly, reorganization of the telephone plan abstracts the step of dialing a number into separate plans for dial and touch-tone phones.

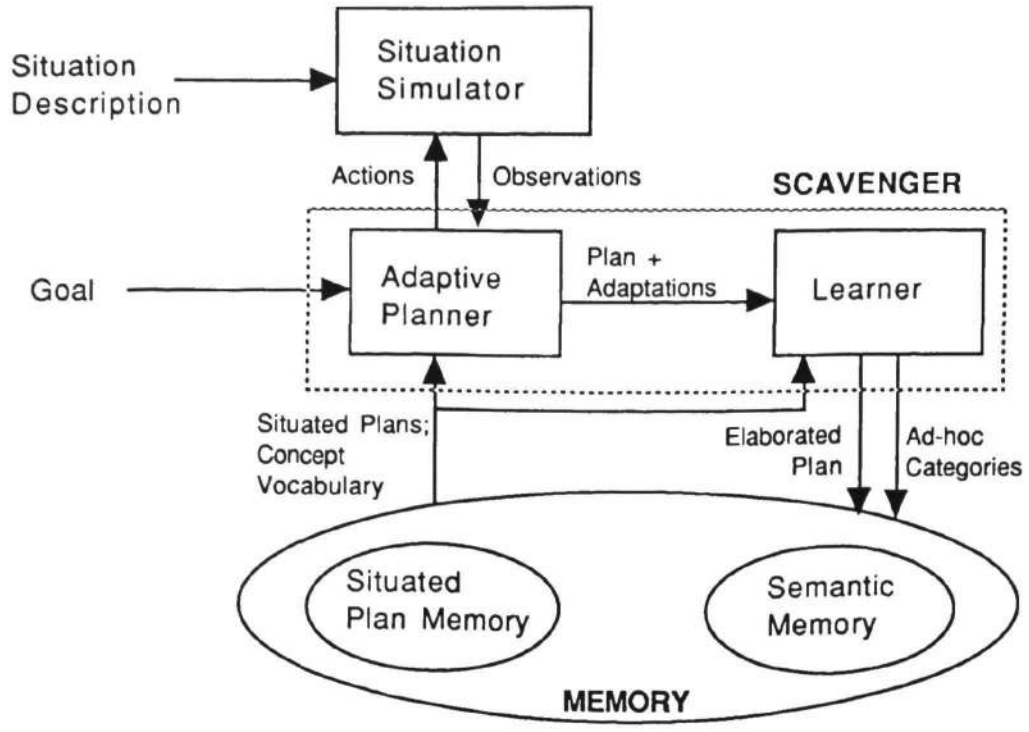


Figure 1: SCAVENGER system diagram

3 An Example of Ad-Hoc Learning

SCAVENGER is given the goal of placing a telephone call in a situation containing the following items: a touch-tone telephone and a personal computer. In predicate-calculus notation:

```
exist(tt_telephone)
parts_of(tt_telephone,[keyboard,telephone_rcvr,
  flexible_cord,receiver_cradle]),
exist(apple_computer)
parts_of(apple_computer,[keyboard,screen,
  slot,on_off_switch]),
exist(desired_number, value,[7,3,6,2,7,0,3])
```

The plan found in memory for this goal is **telephone_plan**, which presumes a dial phone. Part of **telephone_plan** is shown below. The first clause indicates that to make a call one needs to be at a telephone, pick up the receiver, check dial tone, dial, wait for the ring, and wait for an answer. The **dial** step is further decomposed. For each step, the expected preconditions (**prec**) and outcomes (**outc**) are declared. **Role** declarations specify the expected types of the items referenced by a step.

```
steps(telephone_plan,[pick_up_rcvr,
  hear_dial_tone,dial,wait_for_ring,answer,talk]),
role(telephone_plan,desired_number,
  telephone_number),

prec(pick_up_rcvr,[exist,dial_telephone]),
prec(pick_up_rcvr,[exist,telephone_rcvr]),
outc(pick_up_rcvr,[hear,dial_tone]),
reason(pick_up_rcvr,dial),
step_type(hear_dial_tone,observation),
prec(hear_dial_tone,[hear,dial_tone]),

steps(dial,[find_first_digit,select_digit,
  find_digit_on_dial,dial_one_digit,
  hear_clicks,dial_loop_test]),
prec(dial,[hear,dial_tone]),
outc(dial,[have,dialed]),
-- remaining steps and substeps omitted --
```

SCAVENGER has background knowledge about dial telephones, and a small taxonomy of sounds:

```
parts_of(dial_telephone,[dial,telephone_rcvr,
  flexible_cord,receiver_cradle]),
index(dial_telephone,telephone_rcvr),
index(dial_telephone,dial),

isa(tone, sound),
isa(clicking, sound)
```

4 Plan Adaptation

Adaptation of **telephone_plan** proceeds as follows. Execution begins; the first situation difference detected is a failing precondition **exist(dial_telephone)** in step **pick_up_rcvr**. SCAVENGER adapts this step by finding a substitute for **dial_telephone** which exists in the current situation, namely, **tt_telephone**, and modifying the step to use it. The substitution is justified by similarity of type and features.

Another substitution occurs later in the example, where SCAVENGER needs to re-interpret the situation to account for a failing outcome. The step **hear_clicks** represents the act of confirming that a digit has been transmitted (dialed) by listening for the clicking sound in the receiver. In this situation a tone is heard instead. The system justifies the substitution of TONE for CLICKING on the basis that both features are of type SOUND, but more importantly, by the temporal relation of the sound and the dialing action.

Similarity comparisons are based on class relations, as in **hear_clicks**, plus similarity of features, as in **pick_up_rcvr**. Similarity by class is measured by the distance of the items being compared along ISA relations, if present. Feature similarity is based on the features (for example, PARTS) associated with the desired item. Missing or excess features are not counted. In terms of Tversky's model of similarity[Tversky, 1977],

$$S(Sample, Pattern) = F(Sample \cup Pattern)$$

Our F allots extra weight to salient (causally relevant) features. Also, we do not require that features match exactly, so that a simple set-union model is inappropriate. The value v_i contributed to a match by pattern feature i is 1 if there exists a sample feature matching i; else it is the similarity of the best-matching sample feature. The similarity of any pair of features is computed by recursive application of the similarity function. Since recursive similarity computations are computationally explosive, each comparison is given a *search horizon* limiting the level of detail the comparison will explore. The horizon is reduced with each recursion. (Better yet would be horizons based on how significant the result might be in the context of the overall match.)

The substitution of **tt_telephone** for **dial_telephone**, for example, is justified primarily by the presence of a particular salient part, the **telephone_rcvr**. Salience is currently marked explicitly in the knowledge base by clauses of the form **index**(*object, feature*). The relevant knowledge clauses are:

```
parts_of(dial_telephone,[dial,telephone_rcvr,
  flexible_cord,receiver_cradle]),
index(dial_telephone,telephone_rcvr),
index(dial_telephone,dial)
```

Plus the observations about the touch-tone telephone present in this situation:

```
exist(tt_telephone),
parts_of(tt_telephone,[keyboard,telephone_rcvr,
  flexible_cord,receiver_cradle])
```

The features marked salient with **index** can be computed by examining the **telephone_plan** for parts and features that appear in preconditions of steps. That yields the list:

```
prec(pick_up_rcvr,[exist,dial_telephone]),
prec(pick_up_rcvr,[exist,telephone_rcvr]),
prec(dial,[hear,dial_tone]),
prec(select_digit,[exist,desired_number]),
prec(find_digit_on_dial,[exist,dial]),
prec(hear_clicks,[hear,clicking]),
prec(answer,[hear,ringing])
```

Intersecting this list with the static features of the telephone suggests **telephone_rcvr** and **dial** as the salient features of a telephone. Measured this way, salience of features is relative to the system's current state of knowledge.

5 Elaboration of Situated Plans

During adaptation, SCAVENGER notes each adaptation made and the reason – the situation difference – that prompted it. **Telephone_plan** is then *elaborated* to incorporate SCAVENGER's touch-tone phone experience. Elaboration conditionalizes a plan to recognize a new situation and make the appropriate modifications. Elaboration steps through SCAVENGER's memory of an episode, adding the modifications made (step insertions, deletions, and reorderings, and new bindings) to long-term memory. It inserts a *discrimination point* into the plan for each adaptation, specifying the change to be made and describing the *context* in which it applies. A discrimination point is activated when its context matches the situation of execution.

A context is a set of features characterizing a situation.[1] All the discrimination points created by a given learning episode share a common context. The context for the example situation is [**using(telephone), has_part(keyboard)**]. (We use

[1]The context is currently provided to the system as part of each situation. Preferably, the system should abstract out its own context features. This is an area of current research.

the term *keyboard* rather than *touchpad* to emphasize that the system does not need to distinguish the two; they are both "objects with arrays of buttons.") The latter condition discriminates this situation from the known one having context **[using(telephone), has_part(dial)]**.

The clauses added by this elaboration are:[2]

```
step_binding(pick_up_rcvr,
  [isa, telephone, has_part, keyboard],
  [dial_telephone, tt_telephone])
plan_variation(dial,
  [isa, telephone, has_part, keyboard],
  substitute_step, [find_key_on_keypad, for,
  find_digit_on_dial, of, dial, ...])
plan_variation(dial,
  [isa, telephone, has_part, keyboard],
  substitute_step, [dial_one_tt_digit, for,
  dial_one_digit, of, dial, ...])
step_binding(hear_clicks,
  [isa, telephone, has_part, keyboard],
  [clicking, tone])
```

The two **step_bindings** arise from the feature-substitutions discussed in section 4. The first **plan_variation** represents the substitution of a step for finding a desired key on the keypad for that of finding one on a dial; the second represents the substitution of the physical process of pressing a key for that of dialing a digit.

Elaboration also reifies similarity-based associations as *ad-hoc categories*[Barsalou, 1983] making them available as guides for future adaptation. In our example, a new category **dial_telephone_5** is created and marked as subsuming **dial_telephone** and **tt_telephone**. We now have available the notion that dial and touch_tone telephones are intrinsically "similar" (more precisely, *functionally* similar in some set of situations) and we can use this fact in future action and adaptation. Linkages (PURPOSE and ISA*) are set up between the new concept and its specializations to encourage such adaptations to occur. **Dial_telephone_5** approximates the basic-level concept "telephone". Expressed as clauses:

```
isa(dial_telephone,dial_telephone_5),
purpose(dial_telephone,dial_telephone_5),
isa(tt_telephone,dial_telephone_5),
isa_star(tt_telephone,dial_telephone_5)
```

6 Reorganization of Memory

Elaboration makes no concerted effort to integrate discrimination points into memory. As knowledge accumulates, memory needs to be re-organized (cf. Dynamic Memory[Schank, 1982]). In models such as those of Kolodner[1983] and Lebowitz[1987], re-organization is the process that structures knowledge; it is an essential aspect of acquisition. Our re-organization is driven by the desire for improved access to data already structured by experience and the need to identify pertinent details linking situations with actions. It consists of local, syntactic modifications to the plan structure that reduce its complexity, as measured by the number of decisions that need to be made in executing the plan in a given circumstance. Through many such re-organizations, knowledge becomes integrated into memory, and drifts toward routines specialized to specific situations. This is in contrast to models such as that of Murray and Porter[1989], where integration is primarily concerned with maintaining the logical consistency of memory and occurs in a single operation.

The information that drives re-organization is:

- Where (at what steps) run-time decisions occur.
- How many run-time decisions are associated with a given plan and its immediate substeps.
- Which steps are *substantially changed*, meaning that more than 1/3 of their substeps[3] would be modified given the specified context.

For each substantially changed step, we create a new step with the changes "built in," plus a new discrimination point that substitutes the new step for the old one in the relevant context. This operation replaces several decisions with a single one, reducing the number of decisions to be considered at run-time for situations matching the given context.

The **dial** plan experiences a substantial number of modifications in the touch-tone telephone situation: of 6 steps, 3 are affected. Therefore the re-organizer creates a new step **dial_8** and adds a discrimination point to **dial** such that **dial_8** will replace **dial** whenever the current context matches the touch-tone one. **Dial** and **dial_8** share substeps, meaning that later generalizations of these substeps will also be shared. The creation of **dial_8** and conditionalization of **dial** in effect replace **dial** with an abstracted notion of dialing that subsumes both **dial** and **dial_8**. A **steps** clause defines the new step **dial_8**, and a **plan_variation** clause establishes the relationship between the new step and the old step **dial**:

[2] The formats for these clauses are:
step_binding(*stepname,context,*[*identifier_bound,value*])
plan_variation(*stepname,context,type,description*).

[3] This percentage was chosen heuristically; the particular value is not critical.

```
steps(dial_8,[find_first_digit,select_digit,
  find_key_on_keypad, dial_one_tt_digit,
  hear_clicks, dial_loop_test]
plan_variation(wildcard,
  [isa, telephone, has_part, keyboard],
  substitute_step, [dial_8,for,dial,of,*,
  reason, [split, dial, on_context,
    [isa, telephone, has_part, keyboard]]])
```

SCAVENGER's plans are hierarchically structured; each node either represents a primitive action or is decomposed into subnodes. Re-organization merges a set of decision points into a single decision closer to the root of a plan, creating a specialized subplan of wider scope. The context associated with those discrimination points, the description of their situations of applicability, propagates upward, becoming more immediately accessible. Re-organization optimizes plan segments for simplicity of execution. Re-organization should be an experience-driven process, in that the effort will be best spent if the most-used plans receive the most attention. We keep track of how often each step that involves a decision is executed, and periodically reorganize those steps which have been run the most times since the last re-organization.

Other forms of re-organization are desirable. Aggregation [Weld, 1986] is one such form of plan reorganization; we assume that the dialing loop within the **dial** procedure was created by such a method. Others include elimination of redundant steps via detection of causal relevance or irrelevance of steps, elimination of redundant tests by merger or subsumption, and factoring of common subplans or step sequences. Other transformations could promote steps of a subplan into the current plan level, or demote steps, depending on which configuration required the least the number of conditionals. We envision a library of plan-transformation operators of which the above are special cases. These would be realized as a series of pattern-directed plan transformations. These transformations differ from those of Collins[1987] in that they are local, syntax-directed optimizations rather than large-scale transformations requiring an understanding of the overall strategy behind a plan.

7 Summary and Conclusions

We have presented a synthesis of learning, planning, and acting that emphasizes practical action over generality of knowledge and plan efficiency. In commonsense domains one's knowledge is inevitably incomplete. Each new situation has the potential to confound one in new and unexpected ways. Expectations are always subject to disconfirmation by experience. How is a planner to cope with this? Our example illustrates several principles:

The interpretation of a situation is always tentative; every action simultaneously applies and tests that interpretation[Heritage, 1984; Suchman, 1987]. SCAVENGER's success in interpreting the touch-tone telephone as a dial phone not only adjusts the plan, it confirms the system's understanding of the situation and its choice of plan. The correspondence between plan and situation also suggests a generalization that can be made across the two concepts.

Every situation of action is an opportunity for learning. Future calling episodes will expand the system's notion of what a phone can look like, where one is likely to be found, and how it can be expected to behave. Learning is generally ignored in deductive planners, because updating the domain model correctly and consistently is simply too hard.

Working from experience simplifies planning both by suggesting actions and constraining the space of possible actions and interpretations[Hammond, forthcoming; Kolodner and Simpson, 1989; Alterman, 1988; Schank, 1982]. SCAVENGER is able to interpret the touch-tone phone as a kind of dial phone because the dial-phone plan provides an interpretation of the situation that defines the relevant features of a (dial) telephone.

SCAVENGER acquires several types of knowledge. It extends the **telephone_plan** to cope with touch-tone phones, collects information that discriminates situations to which each variant is applicable, and acquires a new category *telephone*.

Knowledge that is genuinely new – not deducible from existing knowledge – can only come from experience. Methods that learn by restating existing knowledge, e.g., *operationalization* in EBL and *indexing* in case-based reasoning, contribute to our understanding of knowledge organization, but are excluded from "learning" in a larger sense. The only escape is to predefine all the basic concepts the system will ever need to know. While this has been proposed as a theory of mind[Fodor, 1975], as an implementation technique it simply defers the problem. The key question is: How is actual learning possible? Although it will probably not take the exact form proposed here, we believe that the concept of ad-hoc learning grounded in experience is a step toward such a method.

References

[Agre and Chapman, 1987] Philip E. Agre and David Chapman. Pengi: An implementation of a theory of activity. In *Proceedings of AAAI-87*, pages 268–272, 1987.

[Agre, 1988] Philip E. Agre. The dynamic structure of everyday life. Technical Report AI-TR 1085, MIT Artificial Intelligence Laboratory, 1988.

[Alterman and Zito-Wolf, forthcoming] Richard Alterman and Roland Zito-Wolf. Planning and understanding: Revisited. In *Proceedings of 1990 AAAI Spring Symposium*, forthcoming.

[Alterman, 1988] Richard Alterman. Adaptive planning. *Cognitive Science Journal*, 12:393–421, 1988.

[Barsalou, 1983] Lawrence W. Barsalou. Ad-hoc categories. *Memory and Cognition*, 11(3):211–227, 1983.

[Chapman, 1987] D. Chapman. Planning for conjunctive goals. *Artificial Intelligence*, 32(3):333–377, 1987.

[Collins, 1987] Gregg C. Collins. Plan creation: Using strategies as blueprints. Technical Report CSD/RR 599, Yale University, 1987.

[DeJong and Mooney, 1986] Gerald DeJong and Raymond Mooney. Explanation-based learning: An alternative view. *Machine Learning*, 1:145–176, 1986.

[Fikes and Nilsson, 1971] Richard E. Fikes and Nils J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3):189–208, 1971.

[Firby, 1987] R. James Firby. An investigation into reactive planning in complex domains. In *Proceedings of AAAI-87*, pages 202–206, 1987.

[Fodor, 1975] Jerry Fodor. *The Language of Thought*. Crowell, 1975.

[Georgeff and Lansky, 1987] Michael P. Georgeff and Amy K. Lansky. Reactive reasoning and planning. In *Proceedings of AAAI-87*, pages 677–682, 1987.

[Hammond, 1986] Kristian J. Hammond. CHEF: A model of case-based planning. In *Proceedings of AAAI-86*, pages 267–271, 1986.

[Hammond, forthcoming] Kristian J. Hammond. Case-based planning: A framework for plannng from experience. *Cognitive Science*, forthcoming.

[Heritage, 1984] John Heritage. *Garfinkel and Ethnomethodology*. Polity Press, Cambridge, England, 1984.

[Kolodner and Simpson, 1989] Janet L. Kolodner and Robert L. Simpson. The MEDIATOR: Analysis of an early case-based problem solver. *Cognitive Science*, 13:507–549, 1989.

[Kolodner, 1983] Janet L. Kolodner. Maintaining organization in a dynamic long-term memory. *Cognitive Science*, 7:243–280, 1983.

[Lebowitz, 1987] Michael Lebowitz. Experiments with incremental concept formation: Unimem. *Machine Learning*, 2:103–138, 1987.

[Minton *et al.*, 1989] S. Minton, J. G. Carbonell, C. A. Knoblock, D. R. Kuokka, O. Etzioni, and Y. Gil. Explanation-based learning: A problem-solving perspective. *Artificial Intelligence*, 40:63–118, 1989.

[Mitchell *et al.*, 1986] T. Mitchell, R. Keller, and S. Kedar-Cabelli. Explanation-based generalization: A unifying view. *Machine Learning*, 1:47–80, 1986.

[Mooney, 1988] Raymond Mooney. A general explanation-based learning mechanism and its application to narrative understanding. Technical Report AITR 88-66, University of Texas at Austin, 1988.

[Murray and Porter, 1989] Kenneth S. Murray and Bruce W. Porter. Controlling search for the consequences of new information during knowledge integration. In *Proceedings of the Sixth International Workshop on Machine Learning*, pages 290–295. Morgan Kaufmann, 1989.

[Pazzani, 1988] Michael J. Pazzani. Learning causal relationships: An integration of empirical and explanation-based learning methods. Technical Report UCLA-AI-88-10, University of California, Los Angeles, 1988.

[Schank, 1982] Roger Schank. *Dynamic Memory*. Cambridge University Press, 1982.

[Suchman, 1987] Lucy A. Suchman. *Plans and Situated Actions*. Cambridge University Press, 1987.

[Tversky, 1977] Amos Tversky. Features of similarity. *Psychological Review*, 84:327–352, 1977.

[Weld, 1986] Daniel S. Weld. The use of aggregation in causal simulation. *Artificial Intelligence*, 30(1):1–34, 1986.

NETWORKS MODELING THE INVOLVEMENT OF THE FRONTAL LOBES IN LEARNING AND PERFORMANCE OF FLEXIBLE MOVEMENT SEQUENCES

Raju S. Bapi and Daniel S. Levine
Department of Mathematics
University of Texas at Arlington
Arlington, TX 76019-9408

Abstract

Networks that model the planning and execution of goal-directed sequences of movements are described, including the involvement of both the prefrontal cortex and the corpus striatum. These networks model behavioral data indicating that frontal damage does not disrupt the learning and performance of an invariant sequence of movements. If the order of performance of the movements is allowed to vary, however, frontal damage markedly reduces ability to perform the sequence.

I: Introduction

The frontal lobes have been implicated in forming strategies for goal-directed behavior (see Nauta, 1971; Fuster, 1980; and Stuss and Benson, 1986 for summaries). This general function seems to involve co-ordination of subsystems that integrate motivational and cognitive information (e. g. Milner, 1964; Pribram, 1961) with other subsystems that link past events or actions across time (Pinto-Hamuy and Linck, 1965; Fuster, 1980, 1985) and anticipate future events or actions (Ingvar, 1985; Gevins et al, 1987).

The motivational-cognitive linkages have previously been simulated in neural networks by Leven and Levine (1987) and Levine and Prueitt (1989). The network architectures used in those simulations were based on principles such as adaptive resonance (Carpenter and Grossberg, 1987) and opponent processing (Grossberg, 1972). In this article, we look at networks that model data of Pinto-Hamuy and Linck (1965) on the performance of movement sequences by frontally damaged monkeys.

The function of learning goal-directed sequences involves classification of spatiotemporal patterns. Hence, based on an idea of Dawes (1989), we combine the adaptive resonance architecture, which classifies spatial patterns, with the avalanche architecture (Grossberg, 1978) which generates sequences. The simple spatiotemporal processing areas would seem not to be located at the prefrontal cortex, but possibly in the corpus striatum. The frontal cortex, we believe, exerts higher-order controls over the functions of this sequence-classifying region. Some of these controls enable formation of complex rules for

sequence classification (see Dehaene and Changeux, 1989, for another model of this process). Others cause biases in the competition between sequence representations, favoring longer over shorter sequences to facilitate attention to a motor plan. Possible architectures for such higher-order controls will be described in Section III.

II: Experimental Data

Some effects of frontal lesions on performance of sequential tasks were studied by Pinto-Hamuy and Linck (1965) in immature macaque monkeys. Postoperative retention was assessed in two types of test: one, in which the subjects were to respond in such a way that they had to push all of several cued panels without repetitions but in any order (internally ordered or flexible sequence test), and another, in which they had to respond in an exact order by pushing a series of panels based on given cues (externally ordered or invariant sequence test). The hypothesis was that frontally damaged subjects would have difficulty retaining or relearning flexible sequences, as it would call for interaction between an internal representation of a sequence and a flexible recall based on previous action performed so as to avoid repetitions. In the case of invariant sequences, on the other hand, the frontals would not have difficulty, as there are sensory cues (lighted panels) available for guiding motor actions).

The experimental results for one subject are illustrated in Figure 1, which shows the percentage of correct responses for one each of the externally and internally ordered sequences both pre- and post-operatively. The subjects were to respond in the internally ordered test situation, for example, if an "O" and a green circle are lighted, by pressing them in either order. In the externally ordered situation, for example, the subjects were to respond by pressing a green circle and a red circle, lighted on a panel, exactly in that order.

Another experiment, by Poppen et al (1965), explored the effects of frontal lesions on sampling and search in human patients, wherein the subjects were to modify their strategies for winning candy, working through a sequence of twenty programs. In each program, only one of the geometric figures displayed randomly on a four-by-four panel set-up is rewarding. These patients had similar difficulties to the frontally lesioned monkeys in maintaining the strategy required to complete the task to criterion. Involvement of the frontal lobes in coordination of movement sequences is also supported by single cell data on the contingent negative variation or expectancy wave (see Fuster, 1985 for discussion). Though this work does not directly suggest neural architectures modeling the capacity for spatiotemporal pattern

processing, it give some evidence of where possible timing controls lie.

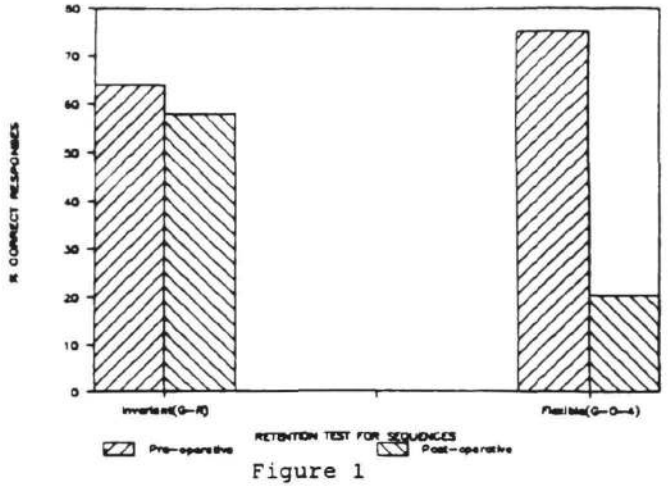


Figure 1

In the next section, we describe possible neural networks that can capture these various events and ways of integrating them. In particular, we discuss methods for qualitatively simulating the results of Pinto-Hamuy and Linck (1965).

III: Network Architectures

First consider how a specific sequence of movements might be encoded in a neural network. One possible mechanism is shown in Figure 2. This architecture was originally developed by Grossberg (1978) and was based on his own previous notion of an avalanche. The avalanche is a network for performance of a ritualistic sequence of motor acts. The network shown here extends the avalanche to include sensitivity to external feedback. The nodes $v_{i,1}$ in that figure are motor reprsentations. The $v_{i,1}$ are active in succession, but external events can alter the exact timing of their firings, or even interrupt the sequence altogether. Hence, the goal-directed actions encoded by the network can be overridden by significant changes in context. Each $v_{i,1}$ has corresponding to it a $v_{i,3}$ to keep it reverberating in short-term memory as long as needed, and a $v_{i,2}$ (influenced by an arousal source) to shut off its reverberation. Each $v_{i,2}$ also activates the next stage $v_{i+1,1}$ of the sequence.

If the sequential performance network of Figure 1 is combined with a two-layer network such as ART (Carpenter and Grossberg, 1987) for coding spatial patterns, the result can be a network for coding spatiotemporal patterns (Figure 3). In Figure 3, each of the $v_{1,i}$ from Figure 2 becomes a node at the category level F_2. The nodes at F_2 learn categories of activity vectors at the input level F_1. During learning, if the input pattern at F_1 mismatches the

pattern of synaptic weights from F_2 to F_1, node A causes short-term memory reset, leading to testing of a new category at F_2.

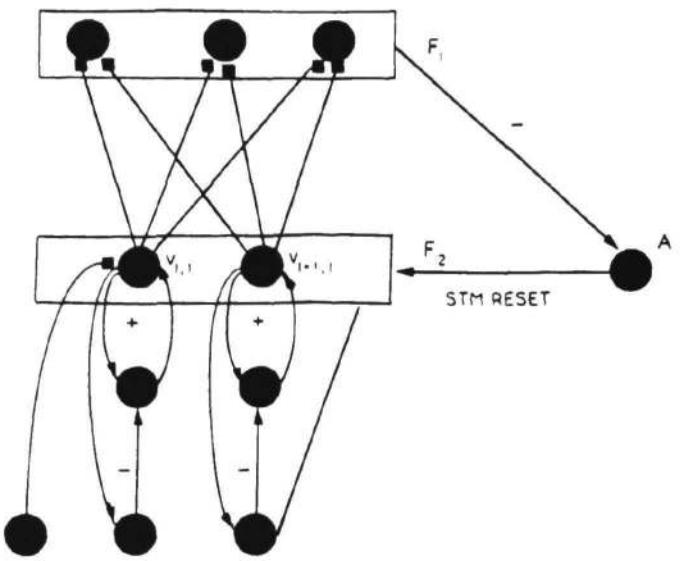


Figure 2

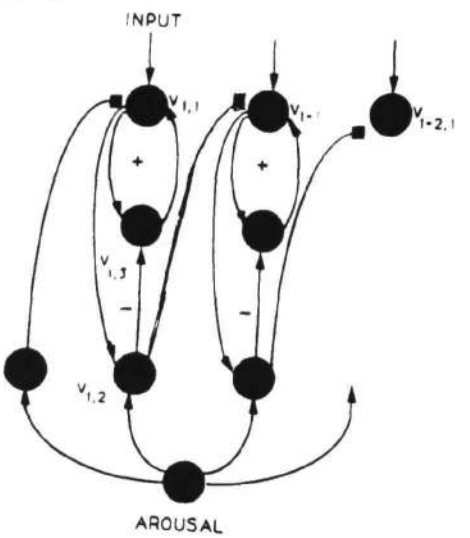


Figure 3

Nigrin (1990) developed another ART-based architecture for coding spatiotemporal patterns, with particular application to speech recognition. His network differs from ours in that it transforms spatiotemporal patterns into purely spatial patterns before encoding them, whereas our network includes the time dimension explicitly.

Recall from above the result of Pinto-Hamuy and Linck (1965) that learning and performance of externally ordered sequences is not disrupted by prefrontal lesions. Hence, if the spatiotemporal categorization network of Figure 3 is analogous to any real brain locus, it is probably elsewhere

than the prefrontal cortex. One good candidate location for such a motor control structure is the corpus striatum, a part of the basal ganglia. Extensive functional connections exist between the prefrontal cortex and corpus striatum (see, e. g., Gerfen, 1989). In the network of Figure 3, such controls could be exerted through the arousal node. This network hypothesis is in line with the general theory (Nauta, 1971; Levine and Prueitt, 1989) that the frontal lobes, through their reciprocal connections with the limbic system and hypothalamus, mediate motivational influences on cognitive and motor functions.

In Leven and Levine (1987) and Levine and Prueitt (1989), aspects of the frontal damage syndrome, namely, perseverative behavior and excessive attraction to novelty, were replicated in neural networks by the reduction of gain from reinforcement signals. A similar decreased signal from the arousal node in Figure 3 could make a motor plan more subject to distraction by interfering events. Indeed, frontal lesions often lead to distractibility (e. g. Grueninger and Pribram, 1969; Wilkins et al, 1987).

But this simple diminution of a reinforcement parameter is far from sufficient to model all effects of the frontal lobes on behavior. We believe that the frontal lobes also exert some higher-order controls on the striatal spatiotemporal categorization field (F_2). We shall now discuss two types of controls on this field. Speculatively, we suggest that these two controls are mediated by the two major functional subdivisions of the prefrontal cortex, the dorsal and orbital regions (cf. Fuster, 1980).

One type of control is a bias in the competition between nodes at the F_2 level of Figure 3. This bias is designed to favor representations of longer sequences over representations of shorter ones, so that if a long sequence of actions is followed by reward, the entire sequence of actions, not just the set of actions close in time to the reward, is likely to be positively reinforced.

An architecture for imposing such a bias on the competition among sequence representations is the masking field, developed by Cohen and Grossberg (1987) and applied to speech recognition by Cohen et al (1987). In speech, for example, representations of an entire word (e. g. MYSELF) tend to dominate representations of parts of that word which are themselves words (e. g. MY, SELF, or ELF). An example of a masking field is shown in Figure 4. In this field, F_1 and F_2 are as in the adaptive resonance networks, but F_1 nodes are interpreted as coding items in sequences and F_2 as coding sequences. In Figure 4(a), connections from F_1 to F_2 grow randomly along positionally sensitive gradients. The nodes in the masking field F_2

grow so that larger item groupings, up to some optimal size, can activate nodes with broader and stronger inhibitory interactions. In Figure 4(b), interactions within F_2 include positive feedback from a node to itself and negative feedback from a node to its neighbors. Long term memory traces at the ends of F_1-to-F_2 pathways adaptively tune the filter defined by these pathways to amplify the F_2 response to item groupings which have previously activated their target F_2 nodes. Tentatively, we propose that sequences represented at the corpus striatum have "copies" at the dorsal frontal cortex which are configured in a masking field.

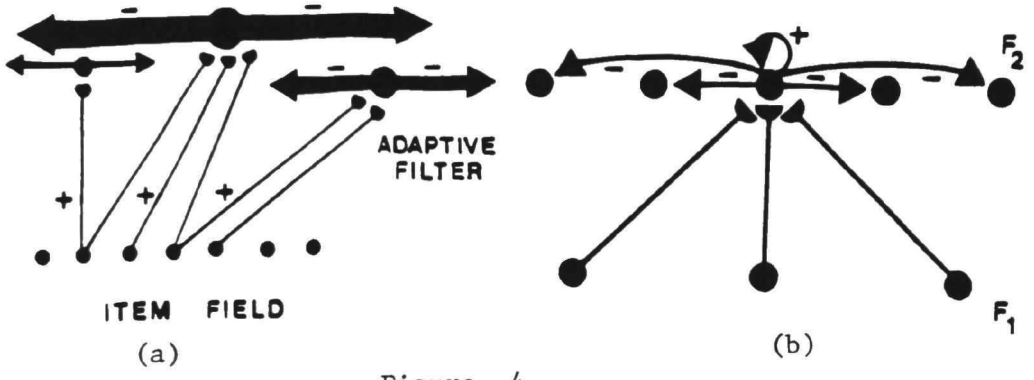


Figure 4

The other type of control that the frontal cortex is likely to exert involves categorization of possible sequences leading to reinforcement. In the case of the internally ordered sequences studied by Pinto-Hamuy and Linck (1965), a mechanism is needed for placing all possible orderings of a sequence in the same category; for example, if the panels to press are a green circle, a letter "O", and a number "4", the sequences GO4, G4O, OG4, O4G, 4GO, and 4OG should be categorized together. This classification must be mediated by the association of all these orderings with a reward. The role of the frontal lobes in such higher-order rule generation has been studied in another model by Dehaene and Changeux (1989).

We suggest that the orbital frontal cortex, through its connections with limbic reinforcement areas, may exert controls which allow many different orderings of the same sequence to be classified together if all orderings are rewarded. Such controls could be mediated by a selective-vigilance criterion for matching sequences to prototypes (see Leven and Levine, 1987, for an example of selective vigilance in an ART network). In current simulations, we are teaching the network several different orderings of the same sequence, then testing it on yet another ordering which may or may not have been presented.

IV: Discussion

The frontal lobes are widely recognized as an important part of a larger control circuit that mediates context-dependent categorizations of both sensory event sequences and motor plans. This circuit also includes parts of the limbic system, basal ganglia, and midbrain, and several monoamine transmitter systems. Foote and Morrison (1987) summarize experimental results on functions of these areas. Hestenes (1990) and several other articles in Levine and Leven (1990) relate these results to possible neural network models of mental illness. Our work herein is part of this larger body of modeling research.

REFERENCES

Carpenter, G. A. & Grossberg, S. (1987). A massively parallel architecture for a self-organizing neural pattern recognition machine. Computer Vision, Graphics, and Image Processing, **37**, 54-115.

Cohen, M. A. & Grossberg, S. (1987). A massively-parallel architecture for learning, recognizing, and predicting multiple groupings of patterned data, Applied Optics **26**, 1866-1891.

Cohen, M. A., Grossberg, S. & Stork, D. (1987). Recent developments in a neural model of real-time speech analysis and synthesis. Proceedings of the First International Conference on Neural Networks, June, 1987.

Dawes, R. E. (1989). The parametric avalanche: Bayesian estimation and control with a neural network architecture. Poster presentation, International Joint Conference on Neural Networks, June, 1989.

Dehaene, S. & Changeux, J.-P. (1989). A simple model of prefrontal cortex function in delayed-response tasks. Journal of Cognitive Neuroscience **1**, 244-261.

Foote, S. W. & Morrison, J. H. (1987). Extrathalamic cortical modulation. Annual Review of Neuroscience **10**, 67-95.

Fuster, J. (1980). The Prefrontal Cortex. New York: Raven. (Reprinted in 1989.)

Fuster, J. M. (1985). The prefrontal cortex: mediator of cross-temporal contingencies. Human Neurobiology, **4**, 169-179.

Gerfen, C. (1989). The neostriatal mosaic: striatal patch-matrix organization is related to cortical lamination. Science **246**, 385-388.

Gevins, A. S., Morgan, N. H., Bressler, S. L., Cutillo, B. A., White, R. M., Illes, J., Greer, D. S., Doyle, J. C. & Zeitlin, G. M. (1987). Human neuroelectric patterns predict performance accuracy. Science, **235**, 580-584.

Grossberg, S. (1972). A neural theory of punishment and avoidance. II. Quantitative theory. Mathematical

Biosciences, 15, 253-285.
Grossberg, S. (1978). A theory of human memory: self-organization and performance of sensory-motor codes, maps, and plans. In R. Rosen and F. Snell (Eds.), Progress in Theoretical Biology (Volume 5). New York: Academic.
Grueninger, W. E. & Pribram, K. H. (1969). Effects of spatial and non-spatial distractors on performance latency of monkeys with frontal lesions. Journal of Comparative and Physiological Psychology, **68**, 203-209.
Hestenes, D. (1990). A neural network theory of manic-depressive illness. In D. S. Levine and S. J. Leven (Eds.), Motivation, Emotion, and Goal Direction in Neural Networks. Hillsdale, NJ: Erlbaum, in press.
Ingvar, D. (1985). Memory of the future: an essay on the temporal organization of conscious awareness. Human Neurobiology, **4**, 124-136.
Leven, S. J. & Levine, D. S. (1987). Effects of reinforcement on knowledge retrieval and evaluation. Proceedings of the First International Conference on Neural Networks, June, 1987.
Levine, D. S. & Prueitt, P. S. (1989). Modeling some effects of frontal lobe damage: novelty and perseveration. Neural Networks, **2**, 103-116.
Levine, D. S. & Leven, S. J., Eds. (1990). Motivation, Emotion, and Goal Direction in Neural Networks. Hillsdale, NJ: Erlbaum, in press.
Milner, B. (1964). Some effects of frontal lobectomy in man. In J. Warren & K. Akert (Eds.), The Frontal Granular Cortex and Behavior. New York: Mc Graw-Hill.
Nauta, W. J. H. (1971). The problem of the frontal lobe: a reinterpretation. Journal of Psychiatric Research, **8**, 167-187.
Nigrin, A. (1990). The real-time classification of temporal sequences with an adaptive resonance circuit. Proceedings of the International Joint Conference on Neural Networks, January, 1990.
Pinto-Hamuy, T. & Linck, P. (1965). Effect of frontal lesions of performance of sequential tasks by monkeys. Experimental Neurology **12**, 96-107.
Poppen, R. L., Pribram, K. H. & Robinson, R. S. (1965). Effects of frontal lobotomy in man on the performance of a multiple choice task. Experimental Neurology **11**, 217-229.
Pribram, K. H. (1961). A further experimental analysis of the behavioral deficit that follows injury to the primate frontal cortex. Experimental Neurology, **3**, 432-466.
Stuss, D. T. & Benson, D. F. (1986). The frontal lobes. New York: Raven.
Wilkins, A. J., Shallice, T. & Mc Carthy, R. (1987). Frontal lesions and sustained attention. Neuropsychologia, **25**, 359-365.

Discovering Grouping Structure in Music

Jacqueline A. Jones, Benjamin O. Miller, and Don L. Scarborough
Department of Computer and Information Science and Department of Psychology
Brooklyn College, City University of New York
jajbc@cunyvm, bombc@cunyvm, dosbc@cunyvm

Abstract

GTSIM, a computer simulation of Lerdahl and Jackendoff's (1983) A Generative Theory of Tonal Music, is a model of human cognition of musical rhythm. GTSIM performs left-to-right, single-pass processing on a symbolic representation of information taken from musical scores. A rule-based component analyzes the grouping structure, which is the division of a piece of music into units like phrases and the combination of these phrases into motives, themes, and the like. The resulting analysis often diverges from the analysis we would produce using our musical intuition; we explore some of the reasons for this. In particular, GTSIM needs to have an algorithm for determining parallel structures in music. We consider alphabet encoding (Deutsch and Feroe, 1981) and discrimination nets (Feigenbaum and Simon, 1984) as algorithms for parallelism.

Introduction

We have been developing a computer simulation of Lerdahl and Jackendoff's (1983) A Generative Theory of Tonal Music (henceforth GTTM) as a model of human cognition of musical rhythm. Our computer simulation, called GTSIM (Jones, Miller & Scarborough, 1988) is a rule-based model with a neural network component. The simulation performs left-to-right single-pass processing on a symbolic representation of information taken from musical scores. Three aspects of music are analyzed: a rule-based component determines metric structure (Miller, Scarborough & Jones, 1988), a neural network determines the tonality, or perceived key, at any point in the score (Scarborough, Miller & Jones, 1989), and another rule-based component determines some aspects of the grouping structure.

We have recently integrated several modules of our model. Now that the modules have been integrated, we are beginning to construct algorithms for their interaction. In particular, we are trying to use strong beats in the metric structure to help find the correct grouping analysis. Grouping analysis is the process by which we divide a piece of music into units like phrases, and then combine these phrases into motives, themes, and the like. While the integrated analysis provides an approximation of the lowest level of grouping boundaries (phrase boundaries) in many cases, the cases for which it fails raise questions about the theory. Recognition of parallelism in music, not yet implemented in our model, seems to be an essential component for producing correct grouping analyses.

Background—GTTM

Lerdahl & Jackendoff's GTTM partitions rhythm into two independent hierarchical components: metric structure and grouping structure. Metric analysis yields a hierarchical representation of metric structure which conforms to

traditional intuitions about meter and accent. The hierarchy represents the strength of the beat at evenly spaced times in the music. Stressed notes (strong beats in the music) correspond to the highest levels of the metric hierarchy. Grouping analysis yields another hierarchy, reflecting intuitions about musical phrases, motives, themes, etc. Grouping preference rules (GPRS) tell us where to find group boundaries, while grouping well-formedness rules tell us how to construct a legal grouping hierarchy from the first level groups. Grouping and metric analysis in GTTM are largely independent, and each can, to a large degree, be carried out without the other. While GTTM's analysis tries to find the best fit between meter and grouping, one cannot be inferred from the other.

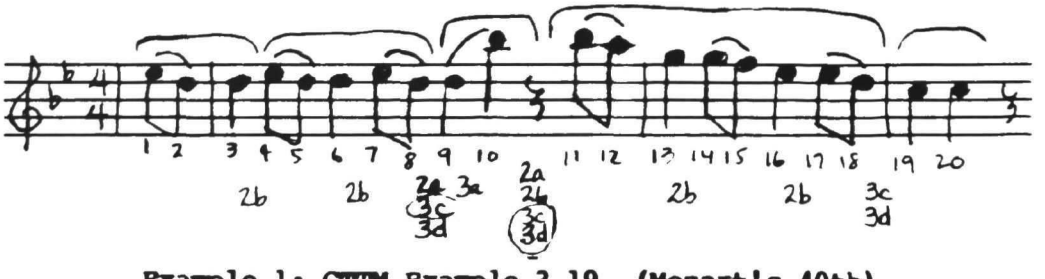
Our Model—GTSIM

We have attempted to devise a model of the process by which a human listens to and understands music. To this end, we have devised a system which processes music from beginning to end, without backtracking. Backtracking - going back and making a second pass through the music once one has heard the entire piece - is not a reasonable model of how humans process music. All our algorithms are constrained by the limits of human memory.

Application of the GTTM Grouping Rules

The grouping module of GTSIM identifies potential grouping boundaries in the score, based on proximity of note onsets or offsets and on significant differences in such attributes as pitch, duration, and articulation (defined in Lerdahl & Jackendoff's grouping preference rules (GPRs) 2 and 3, and their subrules). It places a marker between two notes if there is an application of a rule at that point. The transition point thus marked is a candidate for being an actual group boundary. Our module has successfully marked the rule applications which Lerdahl & Jackendoff find in their own examples. However, we also tend to find spurious candidate boundaries as a result of rigorous application of the rules.

Example 1 shows our initial grouping analysis of the melody at the beginning of Mozart's 40th Symphony, Lerdahl & Jackendoff's Example 3.19. The rule applications which we find and that they do not are circled. In all examples, rule applications are shown below the score, and the groups determined by algorithm INTEG2, described below, are marked above the score.



Example 1: GTTM Example 3.19 (Mozart's 40th)

The three extra boundaries come about through rigorous interpretation of the slur-rest rule, the articulation rule, and the duration rule. We interpret the 8-9 and 10-11 transitions as being boundaries between notes of different articulation, since the notes are not slurred together, and since notes 9 and

11 must be articulated at their onset just like any other non-slurred note. The duration rule applies at 10-11 as well, since notes 9 and 10 are the same length, as are notes 11 and 12, while notes 10 and 11 are of different lengths.

Choosing First Level Boundaries

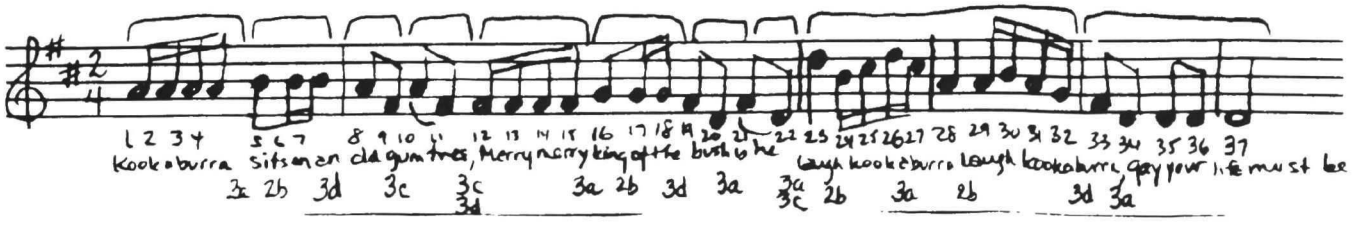
Marking all the rule applications is only the first step in creating the grouping hierarchy. Once the candidate boundaries have been identified by the rule applications, we must decide which of the candidate boundaries are the actual boundaries between groups (phrases). These boundaries divide the piece into the groups that constitute the first level of the grouping hierarchy. Next, we must begin to combine these groups into ever larger units--groups of groups--limited in principle only by the length of the piece of music itself.

Not all of the transitions marked as candidate boundaries can be actual group boundaries. In Example 1, selecting the boundaries at transitions 8-9, 9-10, and 10-11 would violate the GTTM rule that no group should consist of one note. Furthermore, a grouping structure of this sort would violate our musical intuition of how this piece is grouped. In other pieces, there are candidate boundaries between 7 (or more) notes in a row. "Greensleeves," for example, has candidate boundaries at almost every transition (see discussion below). It is not possible for each of these transitions to mark a new phrase in the music.

Our initial attempt to select first level group boundaries used a simple counting algorithm. The algorithm counts the candidate boundaries at each point; the candidate boundaries with more rule applications are selected as actual boundaries; those with the most rule applications are considered larger-level boundaries. The algorithm works well for some pieces; for "Row, Row, Row Your Boat," it nicely divides the piece into four lines with a major break after line 2; for other pieces, it produces no groups at all (some pieces have no transitions with more than one rule application).

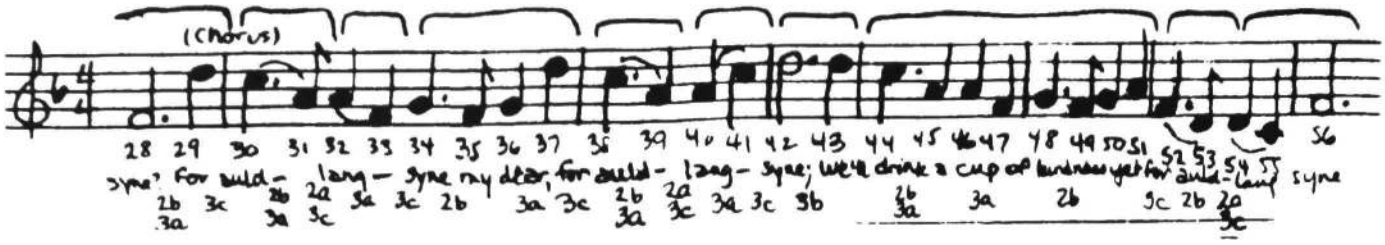
Two more recent algorithms, INTEG1 and INTEG2, are based on the integrated metric and grouping analysis. One of GTTM's metric preference rules says to prefer a metric analysis in which the first note in a group falls on a strong beat. INTEG1 looks at candidate boundaries with more than one GTTM rule application and checks to see whether the note that would begin the group so delineated falls on a strong beat in the metric hierarchy. INTEG1 also produces uneven results; first, many pieces have no transitions with more than one rule application; and second, sometimes a single application of GPR 2 (which identifies rests, among other things) outweighs many applications of GPR 3.

INTEG2 provides better results. All notes which seem to begin groups, based on the fact that they follow one or more rule applications, are checked to see whether they are at a higher metric level than the two adjacent notes. If they are, they are considered to be first level group boundaries. Example 2 illustrates a successful analysis.



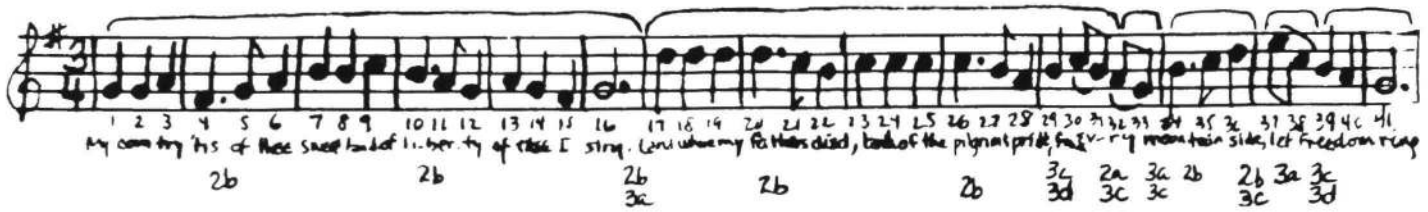
Example 2: Kookaburra

INTEG2 ensures that we will capture only groups which begin on primary or secondary strong beats in a bar; phrases that begin on upbeats, such as those in "Farmer in the Dell" and "Auld Lang Syne" (Example 3, transitions 28-29, 36-37, 42-43, and 50-51) will be ignored.



Example 3: Auld Lang Syne (chorus)

The groups which are established by INTEG2 are not always those which we would pick by looking at the score or by listening to the music. Often the groups are of irregular size throughout a piece; one section will be completely undivided, while another section will be overly divided into many small groups.



Example 4: America

Partly this reflects the fact that we have not yet implemented all the grouping preference rules; one of the significant markers of phrasing in music is parallelism, yet we have not yet attempted to implement GTTM's parallelism rule (see discussion below).

Higher Level Boundaries

Eventually, all the first level groups must be combined into a single well-formed hierarchy of groups. This will be carried out by imposing the higher level rules on the evidence accumulated by the lower level rules. We have attempted to do this only with the first simple counting algorithm.

Discussion

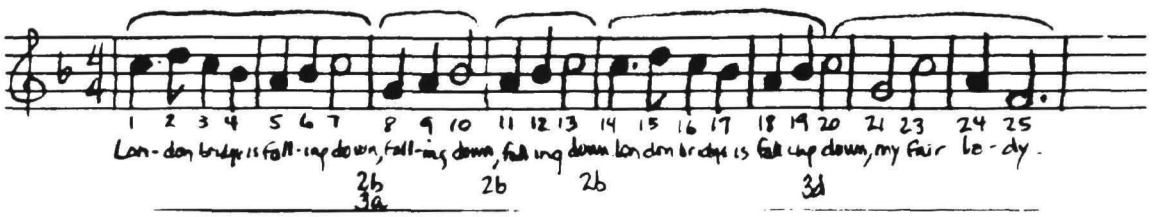
Our attempt to develop a computational model of this formal theory has brought us to a clearer understanding of the limitations of the theory. We also have more awareness of the difficulties in modelling the theory. There are problems at each of the levels of analysis.

Rule Application Level

One problem at this first level is that strict interpretation of the rules causes GTSIM to find excessive numbers of rule applications. Automated application of the rules finds duration differences, as in Example 1, transition 10-11, which Lerdahl & Jackendoff ignore, seemingly because it crosses a rest, an issue unaddressed by GTTM. In other pieces, like "Auld Lang Syne" (Example 3),

"We Three Kings," and "America" (Example 4), there are sections where almost every note has one or more rule applications. Music which alternates long and short notes, particularly music with dotted rhythms, tends to produce many extra attack-point rule applications (GPR 2b). Music that has melodic pitch skips will produce extra pitch rule applications (GPR3a), and music that alternates between slur and standard articulation will have extra articulation (GPR 3c) and slur-rest (GPR 2a) rule applications. Most of these rule applications ought to be ignored but, once produced, must be processed.

A second problem is that other clearly heard boundaries are excluded by the GTTM rules. For example, in "London Bridge," transition 20-21 is not an application of the pitch rule, although a listener feels that it ought to be, because the change in pitch from notes 20 to 21 is the same as the change in pitch from notes 21 to 22. If note 22 were even a half-step lower, 20-21 would be marked as a pitch boundary. The rule correctly rejects the 21-22 transition for the same reason (it is the same pitch difference as 20-21), but this still does not seem satisfactory.



Example 5: London Bridge

A third problem is that some of the folksongs we have used as sample scores do not have clear grouping boundaries to the eye or ear. We intuitively assume that the music divides at commas, or at the ends of lines, but there is often no evidence other than linguistic for finding group boundaries at these points.

On closer inspection, the music sometimes does provide evidence of grouping at such points. Often the music contains parallel sequences which correspond to the linguistic divisions. Two sequences can be said to be parallel if they instantiate the same pattern. A melodic figure and its verbatim repeat are certainly parallel sequences, but parallelism is not necessarily limited to identity. Two musical sequences can be considered parallel if they are similar in rhythm, pitch contour, or internal grouping, or if one embellishes or simplifies the other without changing its essential melodic or rhythmic characteristics. GTTM has a parallelism rule, which says that parallel segments should be combined into parallel parts of higher level groups. However, it does not specify how to pick out the parallel groups and mark them in such a way that they can later be combined into larger groups.

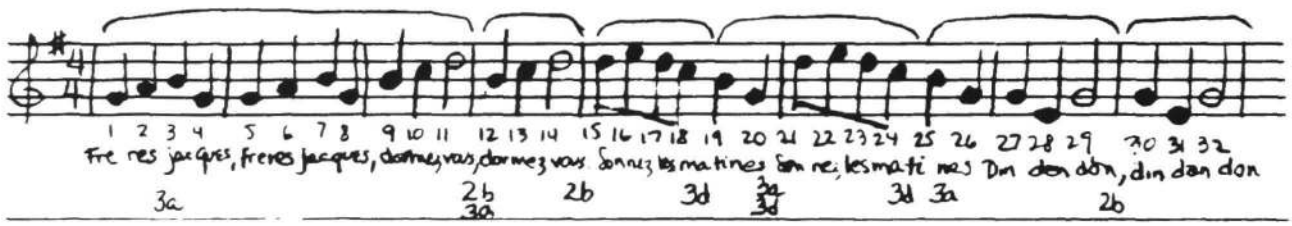
Application of a parallelism rule would find other candidate boundaries, and often these would coincide with the linguistic boundaries. Examples of this can be seen in the first four bars of "America" (Example 4), where there are only two candidate boundaries, neither of which reflects our perception of the actual grouping. However, the first six notes are clearly melodically and metrically parallel to the second six notes; a parallelism rule would find boundaries at transitions 6-7 and 12-13. An algorithm which discovered this parallelism would greatly enhance the analysis. Similarly, the first eight notes of "Freres Jacques" (Example 6) form two parallel groups that are not delineated in any way by the existing rules.

Selecting Group Boundaries Level

Selecting the appropriate first level boundaries is not as simple in a computational model as in a formal theory. The formal model explains the concept, but doesn't need to worry about reasonable implementation of that concept. Thus Lerdahl & Jackendoff's discussion of the selection process involves backtracking, weighting of rules, and determination of parallelism, all of which are computational problems.

Backtracking--applying rules in retrospect, after hearing the entire piece--is not a reasonable model of how humans process music. In their explanation of Example 1, Lerdahl & Jackendoff reject the boundary at transition 9-10 by noting that the boundary at transition 10-11 divides the example in half, and is therefore correct; since there cannot be a group consisting of one note, the 9-10 boundary is incorrect. As a listening model, this assumes that one can listen ahead to the end of the section, determine where the section ends, and then backtrack and decide that the 10-11 boundary marks its division into two large groups. This much backtracking, which requires storing 20 notes in memory, seems unlikely as a model of human performance.

Another problem is selection of first level group boundaries when there is a conflict. Even one rule application, of the right kind, can indicate a true boundary; the conjunction of many rule applications is also strong evidence. However, without some weighting, this decision process cannot be automated. Lerdahl & Jackendoff suggest using a system of weights applied to the rules, where GPR 2 outweighs GPR 3, except when GPR 3 measures a change in dynamics. They do not specify the weighting further. Thus they apply the rules in a rather ad hoc fashion: in Example 1, two rule applications at transition 8-9 are ignored in favor of two at transition 10-11. In other of our own examples, a transition with one rule application clearly (to the listener) outweighs a transition with two or more rule applications. In "Freres Jacques" (Example 6), the boundary that divides the piece in half (transition 14-15) has only one rule application (GPR 2b), while the less important transition 11-12 has two rule applications. At other times, that same single rule application must be ignored, as in "America" (Example 4).



Example 6: Freres Jacques

GTTM's Intensification rule (GPR 4), says that a larger-level boundary may be placed where the effects picked out by GPRs 2 and 3 are relatively more pronounced. This also suggests the need for weighting the effects discovered at the rule applications. Although we have rejected unchanging weights, we plan to incorporate dynamically determined weighting into our next algorithm, noting Deliege's (1987) experiments with weighting GTTM's grouping rules.

A final problem is parallelism. Parallelism is represented in GTTM as a higher level rule, that is, a rule by which to form larger groups from smaller groups (GPR 6). Even if there are other rule applications at the ends of groups, it is the perception that one set of notes is parallel to another set

of notes that enables us to select the correct grouping boundaries. Without parallelism, grouping analyses are improperly segmented. By finding out what we can do without parallelism, we have discovered just how potent a psychological argument parallelism is in determining grouping structure.

Recognition of parallelism, however, is a difficult pattern recognition problem. There are several possible approaches. One algorithm will use a modified discrimination net, modeled after EPAM-III, a model of recognition and learning devised by Simon and Feigenbaum (Feigenbaum & Simon, 1984). EPAM-III was developed to learn to recognize strings of symbols such as a sequence of phonetic features, or a letter sequence.

Another model which we expect to use as a guide to recognizing parallel structure is alphabet encoding. Any sequence can be described in terms of an alphabet that contains all the elements that occur in that sequence and a set of operators that describe transitions between elements and groups of elements. We can encode the same sequence differently by using a different alphabet or set of operators. Alphabet-based coding allows us to represent the hierarchical structure of a sequence; in this way, sequences are reduced to coded chunks that are easier to remember and match with other chunks.

Deutsch and Feroe (1981) assume that pitch sequences are stored internally as hierarchical networks, and use alphabet encoding to represent a hierarchy of nested patterns and subpatterns. The idea that music is represented by such structures accounts nicely for the fact that recognition of melodies is not affected by transposition or, within limits, by tempo changes. At a more detailed level, the complexity of a formula can predict how accurately the corresponding musical sequence is perceived (Jones, Maser & Kidd, 1978). While their model is a strong intuitive representation of the concept, it is not obvious how to implement it within the constraints of our model.

Conclusion

Our computer model of Lerdahl & Jackendoff's GTTM calculates preliminary grouping structure. However, without weighting the grouping rules, and without adding a component which recognizes parallelism, an important psychological factor in determining grouping, we will not get accurate results. Since GTTM does not address these issues, we must develop our own algorithms.

Acknowledgements

This research was funded in part by PSC-CUNY grants to Jones and Scarborough, and by an NSF graduate fellowship to Miller.

References

Deliege, I. (1987). Grouping conditions in listening to music: An approach to Lerdahl & Jackendoff's grouping preference rules. Music Perception, 4, 325-360.

Deutsch, D., & Feroe, J. (1981). The internal representation of pitch sequences in tonal music. Psychological Review, 88, 503-522.

Feigenbaum, E. A., & Simon, H. A. (1984). EPAM-like models of recognition and learning. Cognitive Science, 8, 305-336.

Jones, J. A., Miller, B. O., & Scarborough, D. L. (1988). A rule-based expert system for music perception. Behavior Research Methods, Instruments and Computers, 20(2), 255-262.

Jones, M. R., Maser, D. J., & Kidd, G. R. (1978). Rate and structure in memory for auditory patterns. Memory and Cognition, 6, 246-258.

Lerdahl, F., & Jackendoff, R. (1983). A Generative Theory of Tonal Music. Cambridge, MA: MIT Press.

Miller, B. O., Scarborough, D. L., & Jones, J. A. (1988). A model of meter perception in music. Proceedings of the Tenth Annual Conference of the Cognitive Science Society (pp. 717-723). Hillsdale, NJ: Lawrence Erlbaum Associates.

Scarborough, D. L., Jones, J. A., & Miller, B. O. (1988). An expert system for music perception. Proceedings of the First Workshop on Artificial Intelligence and Music, AAAI-88 (pp. 9-19). Menlo Park, CA: American Association for Artificial Intelligence.

Scarborough, D. L., Jones, J. A., & Miller, B. O. (1989). Modelling music cognition: An expert system. Proceedings of The Arts & Technology II: A Symposium (pp. 132-146). New London, CT: Center for Electronic and Digital Sound at Connecticut College.

EXPLANATIONS OF NUTRITIONAL CONCEPTS: ROLE OF CULTURAL AND BIOMEDICAL THEORIES

MALATHI SIVARAMAKRISHNAN
and
VIMLA L. PATEL

Cognitive Studies in Medicine
McGill University, Montreal

ABSTRACT

In this paper, we explore the role of cultural and biomedical theories in the explanations of complex nutritional concepts by lay-people. Subjects were East Indian mothers who are exposed to both the traditional *Ayurdevic* and modern *biomedical* theories of health and disease in their environment. Subjects' explanations of nutritional concepts after being shown pictures of children suffering from childhood nutritional deficiency diseases contained a modified form of concepts from both models of health practice. The number of biomedical concepts used increased with education level. However, detailed examination of knowledge structures showed that the biomedical concepts were interpreted within the traditional theory of Ayurveda rather than modern medical theory, resulting in a weak restructuring of knowledge with education. Thus only the biomedical concepts which are readily interpretable within the pre-existing traditional theory, are used in everyday functioning. Both the "expert" older mothers who practiced traditional nutrition, and the "expert" nutritional consultants, with university degrees in nutrition used similar strategy of forward reasoning in diagnosing the problem. The difference was in the use of knowledge base; the former used a socialized form of Ayuveda and the latter used biomedical knowledge.

INTRODUCTION

Recent studies on scientific understanding and conceptual change has focused on the notion of a lay adult or a child as an intuitive scientist. The scientist explores the environment, constructs mental models of it, and evaluates these models against empirical evidence. Similarly, lay-people make sense of their environment by processing information and constructing and evaluating the models they generate (Kuhn, 1989). The intuitive conceptions of children and lay-adults are constructed on the basis of their everyday experiences and are constrained by their global theory of the world. While some researchers think of these naive knowledge structures unlike scientific theories (DiSessa, 1988), to be fragmented and loosely connected, others believe that they form coherent and systematic sets of ideas akin to scientific theories (McClosky, 1983; Clement, 1983; Wiser & Carey, 1983; Nersessian & Resnick, 1989).

It is of fundamental importance for research in instruction to look at how these naive conceptions based on a global theory are differentiated and restructured on exposure to domain specific theories that occur with formal instruction. While some learning may consist of the acquisition of "totally" new knowledge, most of the learning that occurs in life is either incorporated within prior knowledge or modifies prior knowledge (Vosniadou & Brewer, 1987).

Studies on knowledge restructuring in learning have shown that the process of knowledge acquisition can be characterized in terms of two types of restructuring - a weak restructuring and a strong/radical restructuring. Two successive conceptual systems are structurally different in the weaker sense if the later one represents different conceptual relations than the earlier one, and if the pattern of these relations motivate superordinate concepts in the later system that are not represented in the earlier one (Carey, 1986). A conceptual shift from a novice to an expert has been described as "weak" form of restructuring (Chi et al.,1982; Patel et al., 1988). Two successive conceptual systems are structurally different in the stronger sense if the transition between the two involves conceptual change (Carey, 1986) that is, changes in domain of phenomena to be accounted for by the theory, changes in explanatory mechanisms, and changes in individual concepts. Many researchers have postulated that learning involves knowledge restructuring in the strong sense.

The kind of restructuring that occurs with instruction has implication for planning educational strategies. If the purpose of education is to promote weak restructuring, then we need to build on the existing schema.

However, if learning is viewed as strong restructuring, then the emphasis to be placed on prior knowledge will be different (Vosniodou & Brewer, 1987).

In this paper, we explore the nature of knowledge restructuring that occurs in an everyday reasoning situation. Specifically, we examine lay-peoples' understanding of complex concepts in a naturalistic environment where they are exposed to two types of "knowledge" simultaneously — a traditional indigenous knowledge and a modern school-based, and popular scientific knowledge. The nature of indigenous knowledge is more practical, where social interactions and cultural contexts play a significant role (Ragoff, 1984). The school-induced knowledge is more formal in nature and is embedded within scientific theories.

The question is, what is the nature of the knowledge used in explanations of concepts used in everyday functioning where more than one kind of theory exists simultaneously. What is the role of schooling in knowledge restructuring for everyday practice? Finally, we examine the process by which the mothers reason about their concepts in relation to their environment.

METHODOLOGY

This study explores lay-persons' explanation of complex concepts in the domain of nutrition. Specifically, the research examines the nature of knowledge structures of mothers with different levels of formal schooling in their explanations of complex childhood nutritional concepts, in relation to the traditional and biomedical knowledge.

The subjects were a group of East Indian mothers, born and educated in India, but now living in Canada. These mothers have knowledge of traditional *Hindu Ayurvedic* medicine, which is practiced as an indigenous form of medicine in India. The sample included mothers with no formal education and mothers with university education. This latter group is exposed to more biomedically oriented health concepts. Three expert nutritional graduates in modern practice were used as a comparison group in one aspect of the study. The specific nutritional concepts examined are those within the major nutritional problem among children — Protein Energy Malnutrition (PEM), both chronic and acute (characterized by Marasmus and Kwashiorkar).

Using detailed semi-structured questions, each mother was interviewed in the native language for background information. Next, they were shown pictures of East Indian children showing signs and symptoms of the nutritional deficiency disorders, asked to identify the problem and then explain how and why they occur, and finally, suggest treatment procedures. The symptoms were also verbally explained to promote understanding of the problem. The child with Marasmus showed symptoms of extreme wasting of subcutaneous fat and muscle. The child with Kwashiorkar showed symptoms of misery, generalized edema, enlarged abdomen, skin changes (dry, flaky and peeling), and thin, dry hair on the head. All interviews were audiotaped and transcribed for analysis.

REFERENCE MODELS

Reference models were developed to explicate how the nutritional deficiency concepts are interpreted within the two frameworks. The *biomedical model* interprets a disease in terms of underlying pathophysiology, where each disease is explained individually in terms of physiological and biomedical changes (Shils & Young, 1988). Examples of such a model to account for chronic PEM, Kwashiorkar and Marasmus are given in Figures 1 and 2 respectively. Primary PEM results from insufficient food intake or from the ingestion of food with proteins of poor nutritional quality. When PEM is chronic children show growth retardation. In a severe condition, these can lead to Kwashiorkar (protein deficiency) or Marasmus (energy deficiency). Both causes and treatment are explained by biomedical theory.

The traditional *Ayurvedic model* has a more holistic view of health and disease. Man is viewed as a microcosm of the universe and so the five basic elements (ether, air, fire, water and earth) present in all matter also exist within each individual (Lad, 1985). These elements manifest in the human body as the three basic principles, or humors, known as *tridosha.* According to the Ayurvedic model, disease is viewed as a state of imbalance of the humors and treatment aims to restore the balance. Within this framework, all individuals have a predominance of one of the three humors *(vata, pitta and kapha)* in the body and thus are susceptible to diseases due to the possibility of aggravating the inherent imbalance of the humors. The severity of the disease is dependent on the degree of humoral derangement. Health state is achieved by restoring the balance by providing the "right" foods. The interpretation of the cause and the treatment of disease is always in relation to normal health and they are given in Figure 4 and 5, respectively.

The concepts generated in the subjects' responses were compared with concepts in the two models for their similarity in node-link structures and the interpretation of these structures within each of the two theoretical frameworks.

RESULTS

COMBINED USE OF MODIFIED TRADITIONAL AND BIOMEDICAL CONCEPTS

The percentage of subjects whose concepts matched the biomedical and traditional models varied with education. All subjects' explanations contained a modified version of both biomedical and traditional concepts, forming an independent composite model. However, the subjects with college and university education used a greater number and less modified biomedical concepts than mothers with secondary or no formal education. The biomedical concepts used were related to a few specific concepts, namely insufficient food, food shortage, indigestion, and dietary deficiency.

BIOMEDICAL CONCEPT INTERPRETED WITHIN TRADITIONAL THEORY

Although the mothers' mentioned biomedical concepts, these were interpreted in relation to the traditional theory of health practice. This was found by mapping the concepts to the explanation of concepts given by the subjects. This is illustrated with the following examples.

a) Concept of Maternal Malnutrition: This concept, as it appears in the biomedical model for the cause of chronic PEM, refers to the mothers' poor nutritional status during/prior to pregnancy, which is likely to produce an underweight newborn. When this is compounded after birth by insufficient food intake, it may result in PEM in the child.

According to the Ayurvedic model, however, an individual's constitution is determined at conception, and by the permutations and combinations of the bodily air, fire and water that manifests in the parents bodies. Thus, unlike the biomedical model, the Ayurvedic model emphasizes the importance of heredity factors as a determinant of the child's constitution and therefore, the child's body weight/frame.

Figure 5 gives a structured representation of the protocol of a mother with no formal education. The subject characterizes the child as thin because the mother is thin. The emphasis is on genetic and heredity factors rather than on parental malnutrition. Although the biomedical model recognizes the role of hereditary factors (parental stature) in determining the child's height, and also appreciates the family resemblances in relative weight and body build, hereditary factors do not feature in the causative model for height-weight retardation in the child (cf. Figure 1).

When the mothers talk about the child being thin because the mother or the parents are thin, they seem to be focusing on the concept of inherited constitution rather than the concept of maternal malnutrition. This is further evident when the subject in Figure 5 relates "child having a thin constitution" causally to height/weight retardation in the child. The mothers' lack of concern for the problem by saying "cannot do anything about it" further exemplifies this interpretation of the concept.

b) Concept of Intestinal Parasites/Worms: This concept of worm infection/parasites for the cause of chronic PEM could also be identified in the mothers' explanations. According to the biomedical view, treatment of parasitic infections includes the use of chemotherapeutic agents to destroy the parasites. Although prepurgation tends to eliminate the mucus and the fecal debris that protect the parasite and post-purgation aids in the removal of the killed or anesthesized worms, purgation is needed neither before nor after administration of the drug in most cases of parasitic infections such as hookworm or roundworm. Also the increased peristalsis due to purgation may accelerate the passage of the drug through the intestinal tract. The worm infection may cause diarrhea, in which case purgation may do more harm than good. However, the treatment recommended by the mothers in this study was the administration of medications such as castor oil given at night to cause loose bowel motions in the child to remove the worms. This is illustrated once again in Figure 5. Castor oil is a well known laxative prescribed in Ayurveda. This concept of treatment has a strong bearing on the Ayurvedic concept of elimination therapy, according to which the causative agent needs to be eliminated from the body before any therapy is initiated. It should be noted that this concept of purgation was also common to mothers with formal education.

c) Concepts of Causation and Treatment of PEM: According to the biomedical model, the central concept of causation for both Marasmus and Kwashiorkar is a dietary deficiency of nutrients - predominantly energy deficiency for Marasmus and protein deficiency for Kwashiorkar. Concepts of impaired digestion and absorption, and abdominal distention can be identified as outcomes of the deficient state, with a causal relationship linking the two concepts as seen in Figures1 and 2.

Figures 6 and 7 show the conceptual representations of the mothers' protocols in reasoning about the cause and treatment of Marasmus and Kwashiorkar in the child. In comparing the generated mothers' conceptual networks with the biomedical model, we found that the concept of nutritional deficiency was identified in mothers' explanations. However, the concepts of impaired digestion and absorption, and liver complication are featured as independent causes of the problem for both Marasmus and Kwashiorkar. The mothers lacked the relation between the state of malnutrition and the state of impaired digestion and absorption, or liver complication seen in the biomedical model in Figure 2.

Impaired digestion secondary to a sluggish gastric fire or *Agni* , caused by an imbalance of the humors is central to the causation of all diseases according to the Ayurvedic model (Figure 3). The mothers'

identification of impaired digestion and liver complication as a causative factor for both Kwashiorkar and Marasmus has a striking resemblance to this concept as it appears in the traditional model.

The subjects' treatment for Kwashiorkar and Marasmus parallel this understanding of the cause of the condition and, therefore, deal specifically with the liver complication, thereby ignoring the problem of malnutrition. Medicinal herbs are suggested to treat the liver condition (Figure 8) and a fat-free diet is prescribed (Figure 6) by the mothers. Various drugs have been used in Ayurveda to stimulate the liver and to digest the *ama*, and are used for the treatment of jaundice. A fat-free diet was recommended for jaundice treatment in the biomedical texts prior to 1950, but subsequent studies have shown that fat restriction does not have any effect on the course or outcome of the disease. However, it has been reported that the Ayurvedic physicians or the *Vaidyas* are still following this outmoded biomedical approach for the treatment of jaundice since they and their clients find it effective.

REASONING ABOUT NUTRITIONAL-DEFICIENCY DISORDERS

Four of the older mothers with no formal education identified the problem from the given data by using inductive form of reasoning. The disorder categories identified were based on Ayurvedic principles. For example, in Figure 6 the mother uses "bulging stomach" and lack of general weight gain to diagnose "Jigar and Tilli" disorder. Our expert biomedical nutritional consultants (3), also used the given data to arrive at a diagnosis using inductive reasoning. However, the diagnoses were based on medical categorization.

CONCLUSIONS

The traditional knowledge of Ayurvedic medicine appears to be prominent in all subjects explanations of nutritional-deficiency problems. This traditional model supports the explanation of the nutritional deficiency problem at the descriptive level rather than at a predictive and explanatory level supported by the biomedical model.

With greater exposure to schooling and media, the mothers use terminology similar to biomedical concepts, but their use is compatible with traditional theories rather than with new biomedical theories. The result is that changes in knowledge structures involve a simple addition of concepts and relations to the pre-existing knowledge structure, without changing the theoretical interpretations of the model. This is a weak restructuring of knowledge.

Understanding of prior knowledge of traditional nutritional practices are important if we are to show the benefits of the school system, where biomedical information is presented as new information unrelated to prior beliefs.

REFERENCES

Chi, M.T.H., Feltovich, P.J. & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science, 5,* 121-152.

Clement, M. (1983). A conceptual model discussed by Galileo and used intuitively by physics students. In D. Gentner and A.L. Stevens (Eds.), *Mental Models* (325-339). Hillsdale, NJ: Lawrence Erlbaum.

DiSessa, A. (1982) Unlearning Aristotelian physics: A study of knowledge based learning. *Cognitive Science, 6,* 37-75.

DiSessa, A. (1988). Knowledge in pieces. In G. Forman and P.B. Pufall (Eds.), *Constructionism in the computer age.* Hillsdale, NJ: Lawrence Erlbaum.

Kuhn, D. (1989). Children and adults as intuitive scientists. *Psychological Review, 96:4,* 674-689.

Lad, V. (1985) Ayurveda - The science of self healing. *A Practical Guide.* New Mexico: Lotus Press.

McCloskey, M. (1983). Naive theories of motion. In D. Gentner and A.L. Stevens (Eds.), *Mental Models* (199-324). Hillsdale, NJ: Lawrence Erlbaum.

Nersessian, N.J. & Resnick, L.B. (1989). Comparing historical and intuitive explanations of motion: Does "naive physics" have a structure? *Cognitive Science.*

Patel, V.L., Groen, G.J. & Scott, H.M. (1988). Biomedical knowledge in explanations of clinical problems by medical students. *Medical Education, 22* 398-406.

Ragoff, G. & Lave, J. (1984). Everyday cognition: Its development in social context. Cambridge, MA: Harvard University Press.

Shils, M.E. & Young, V.R. (1988). *Nutrition in health and disease, 7th Edition.* Philadelphia, PA: Lean and Febiger.

Vosniadou, S. & Brewer, W.F.(1987). Theories of knowledge restructuring in development. *Review of Educational Research, 57:1,* 51-67.

Wiser, M. & Carey, S. (1983). When heat and temperature were one. In D. Gentner and A.L. Stevens (Eds.), *Mental Models* (267-297). Hillsdale, NJ: Lawrence Erlbaum.

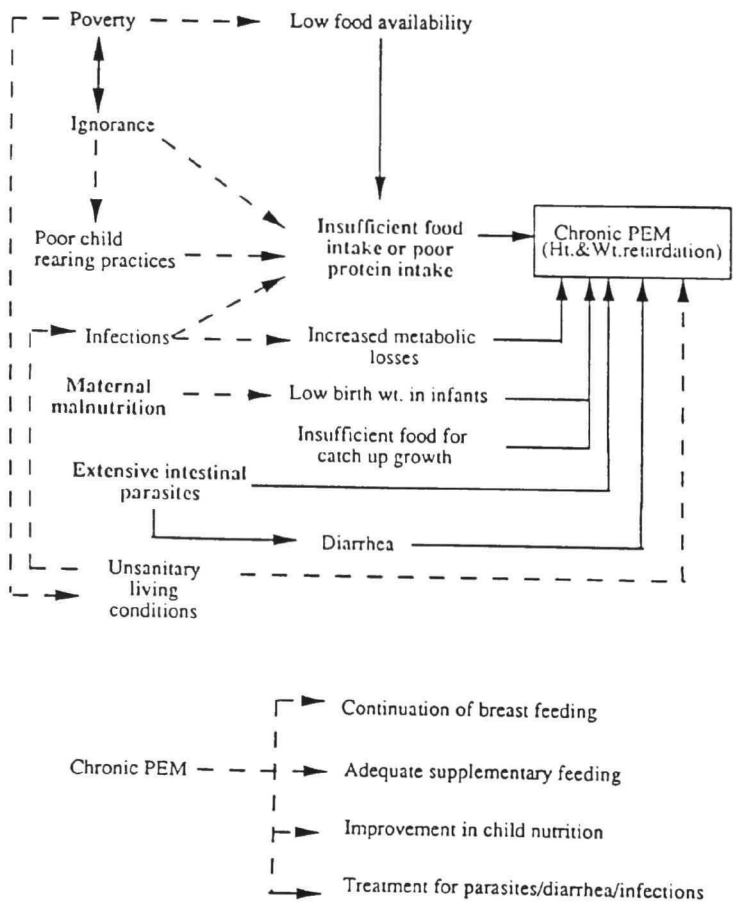


FIGURE 1: BIOMEDICAL MODEL FOR CAUSE AND TREATMENT OF CHRONIC MILD PEM

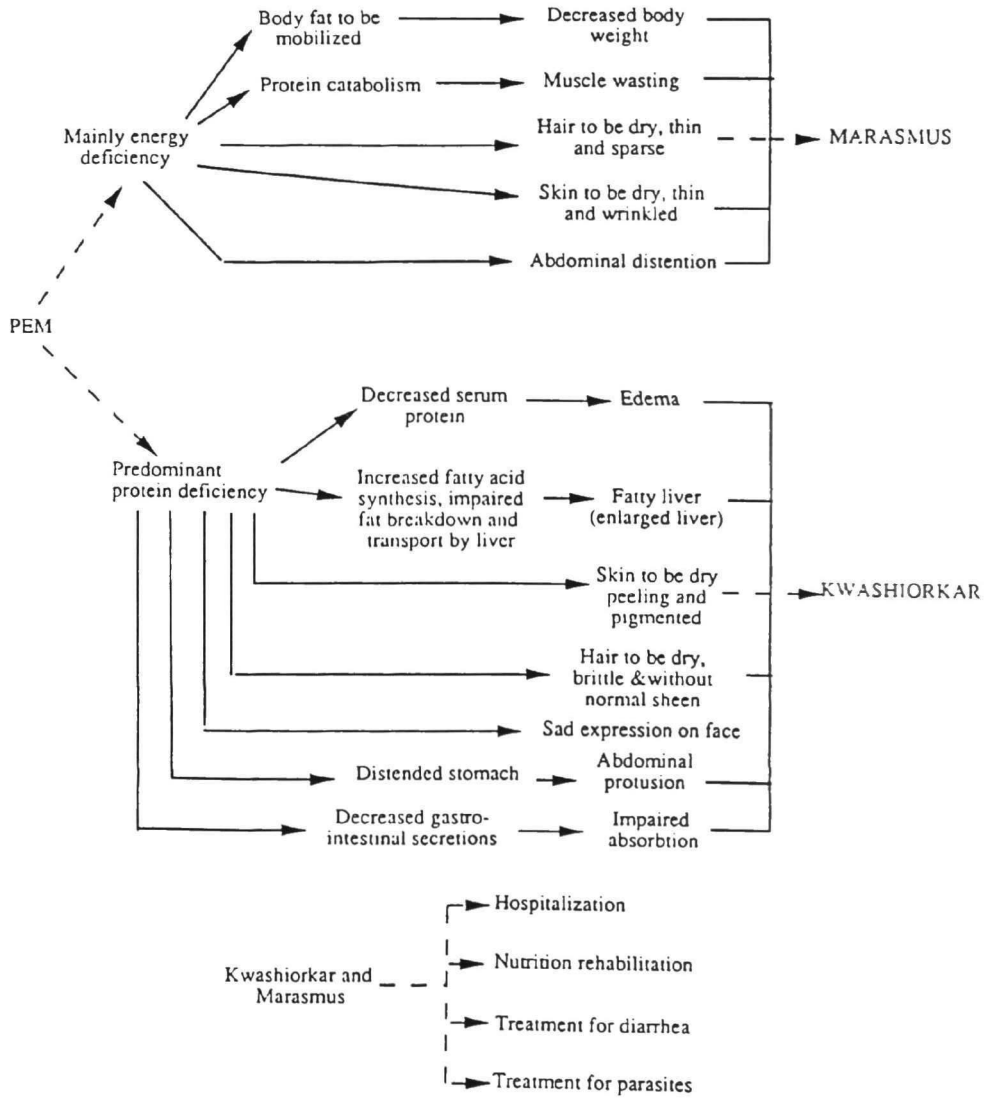


FIGURE 2: BIOMEDICAL MODEL FOR CAUSE AND TREATMENT OF KWASHIORKAR AND MARASMUS

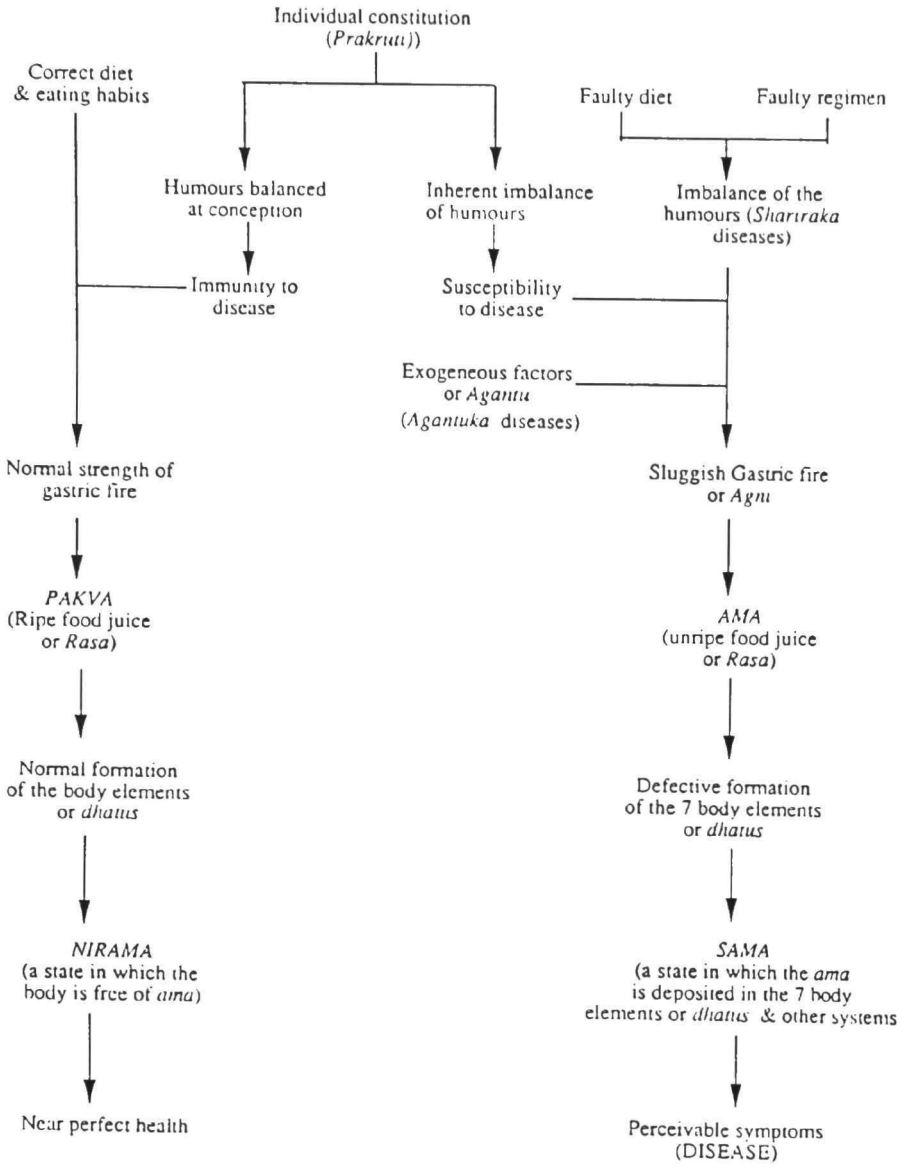


FIGURE 3: THE AYURVEDIC MODEL FOR THE CAUSE OF DISEASE

⟶ Causal relation

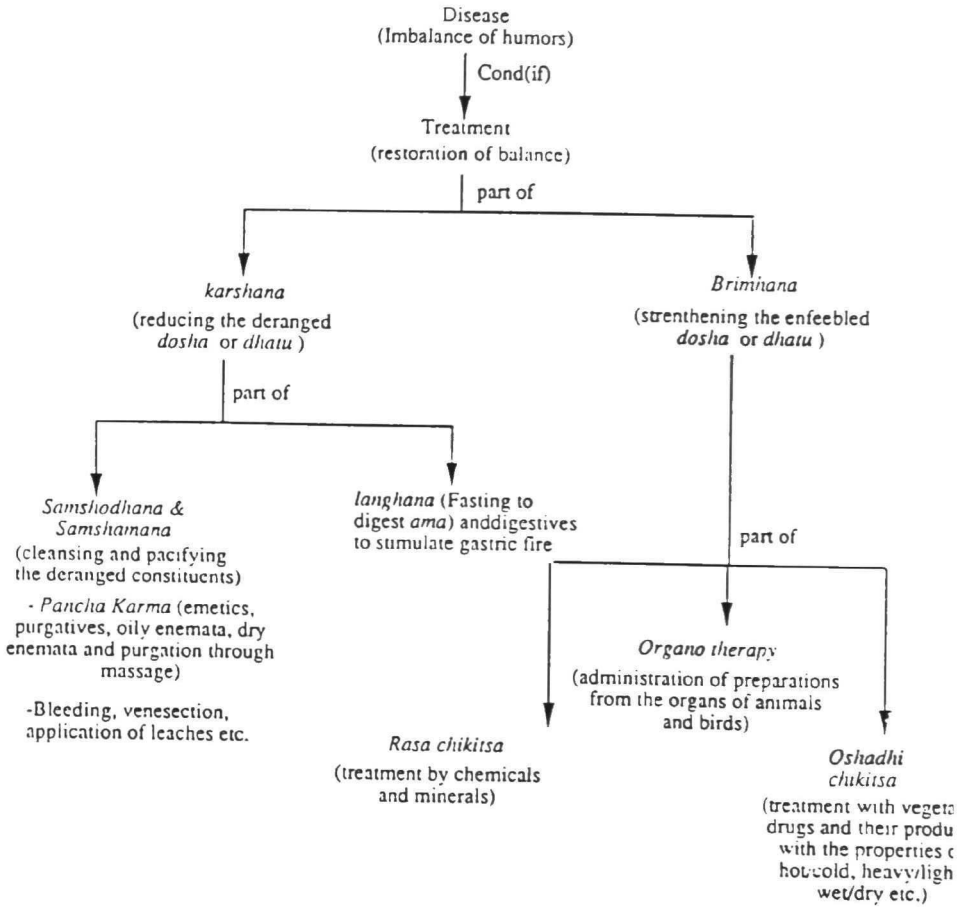


FIGURE 4: CONCEPT OF TREATMENT IN AYURVEDA

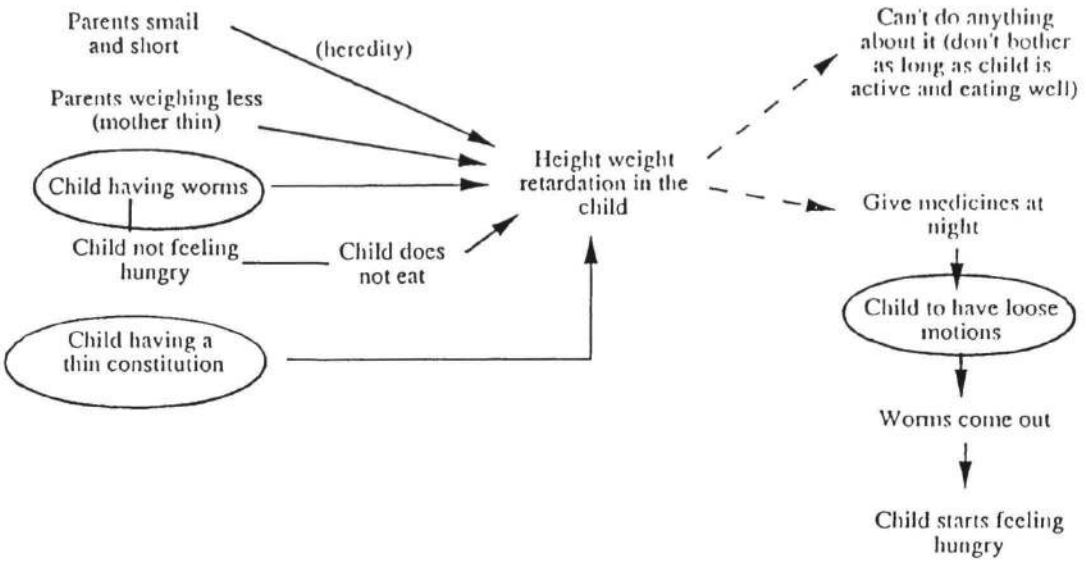


FIGURE 5:Protocol of mother (#4) with no formal education for the cause and treatment of height weight retardation in the child

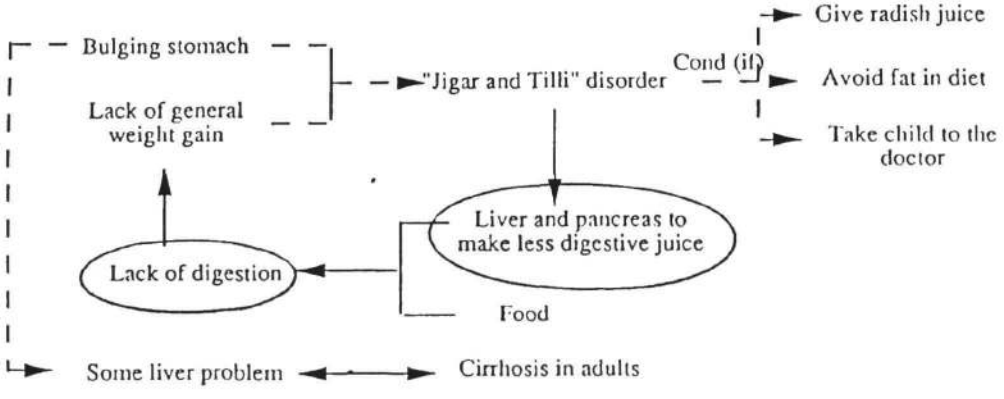


FIGURE 6: REASONING ABOUT CAUSE AND TREATMENT OF KWASHIORKAR GIVEN BY MOTHER(# 4)WITH NO FORMAL EDUCATION

Causal relation
Conditional relation
Associated with

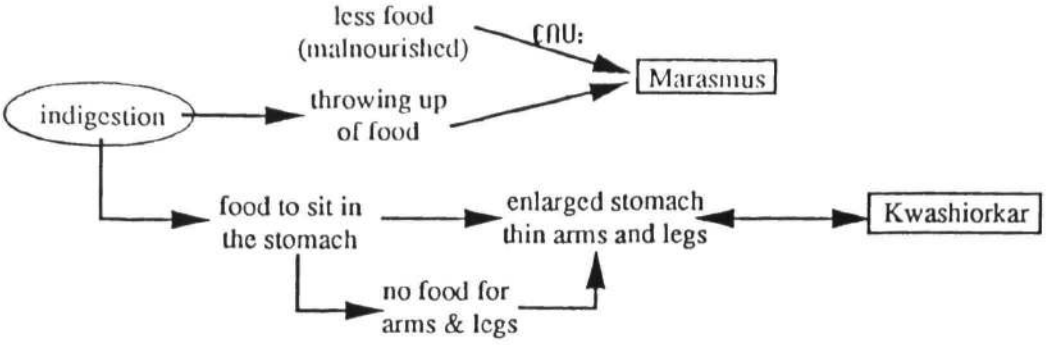


FIGURE 7: Reasoning about causality of Kwashiorkar and Marasmus by a mother (#2) with no formal schooling

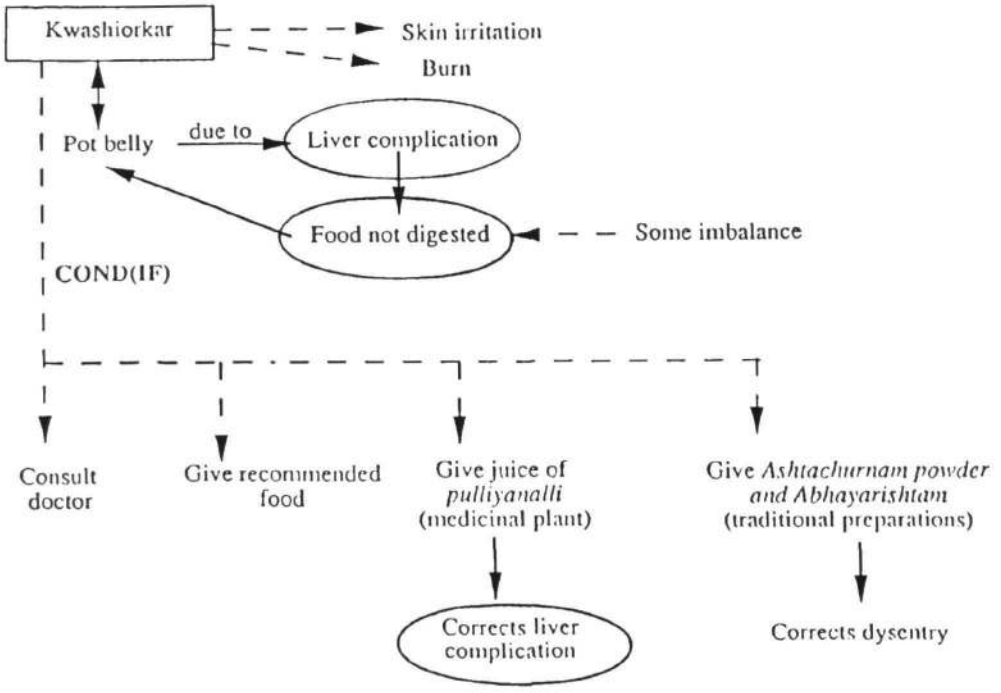
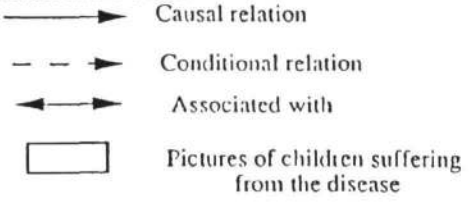


FIGURE 8: Protocol of mother (# 20) with college education for the cause and treatment of Kwashiorkar in the child.



Ill-Structured Problem Solving in Instructional Design

James G. Greeno, Margaret Korpi, Douglas Jackson, and Vera Michalchik
Stanford University

Abstract

This work extends the information-processing theory of problem solving to the domain of ill-structured problems by presenting a framework for understanding instructional design. The design task is dominated by constructive versus search processes. Verbal protocols are analyzed in terms of knowledge and processes used.

Introduction

The information-processing theory of problem solving (Newell & Simon, 1972) has provided a framework for successful analyses of human performance on many well-structured problems (see Greeno & Simon, 1988, for a review). The question of accounting for ill-structured problems within that framework was recognized as a challenge early on (Reitman, 1969; Simon, 1975). Several recent studies of design problem solving (Guindon,1988; Kant,1985; Steler & Newell,1988; Ullman, Dietterich & Stauffer,1988) are contributing to the analysis of ill-structured problems in rigorous terms. Our research, which is coordinated with work by Pirolli and his associates (Goel & Pirolli,1989), contributes to this body of research by studying processes of the design of instructional materials.

The major theoretical idea developed in the information-processing theory of problem solving is the concept of search within a problem space. Problems such as cryptarithmetic and anagrams involve search in a space of possible arrangements; problems such as the Tower of Hanoi or logic proof exercises involves search in a space of possible sequence of operations. Theoretical analyses by Newell and Simon (1972) and others developed and elaborated the concept of the problem space as a set of materials, the set of states that could be produced using the materials, the initial and goal states of the problem, the operators that can be used to transform states, the heuristic methods that the problem solver has available, and the knowledge of the domain that the problem solver uses in constraining search and generating plans.

In contrast, the major focus of attention in a design task is on processes of construction, rather than search. To a considerable extent, the problem solver's task includes construction of the problem space, as well as finding a solution within the problem space. Within the general goals and constraints that are specified, significant components of the problem materials, goals, and constraints are supplied by the problem solver in the process of working on the problem. Of course, well-defined problems include elements of construction (e.g., Greeno, Magone, & Chaiklin,1979; Newell & Simon,1972) and design problems include elements of search, but the balance between construction and search is markedly different.

A Framework for Processes of Design

The account of instructional design problem solving that we have developed uses a framework that seems applicable to a variety of design tasks. We distinguish three attributes of the problem space: subproblems, types of knowledge, and operators. We also distinguish

three components of the solution to the problem: materials that are used; an arrangement of the materials that is constructed; and functions that design components satisfy.

Subproblems involve two different aspects of the task: constructing the design itself, and constructing the problem space. Constructing the design includes determining the materials to be used, determining the arrangement of the materials, and determining the details of implementation. Constructing the problem space includes clarifying the problem, generating goals and constraints, and determining what information is needed to construct the design. The types of knowledge include knowledge about materials in the domain in which the design occurs and knowledge of the genre of objects to which the design belongs. Operators include adding components to the design, and elaborating and modifying components that have been included in the design.

We instantiated this framework for the task of instructional design by developing a scheme for coding verbal protocols obtained while individuals designed instructional materials.

Method

Participants in the study were eight students in the Stanford Teacher Education Program (STEP). Four of the students had recently graduated from the12-month program, and four were new to the program. The participants learned about a fictitious vehicle called the VST2000, and then were asked to design instruction about it. Information about the VST2000 was presented in a computer-based tutorial that simulates the device and presents operating procedures and functional relations among the components. The VST2000 can be run using three alternative sources of energy. The computer display shows a schematic wiring diagram of the vehicle, and includes switches that select among the energy sources, turn components on or off, and establish connections among the components. The simulation has been used in previous research on knowledge that facilitates learning operational procedures, generating novel operating methods, and solving diagnostic problems (Greeno & Berger, 1987; Greeno & Berger, 1990).

In a subsequent session, after participants had learned about the VST2000, each one was asked to design instruction about the vehicle. Two problems were given: (a) designing materials that a teacher could use to instruct high-school-aged students in the operation of the VST2000, and (b) designing materials for a high-school general science course in which the VST2000 would be used as an example of various principles about energy. For the instruction in operations, participants were told to assume that the VST2000 was a real machine and to design a general plan for use in teaching its operation. For the instruction about principles, participants were told to treat the VST2000 as the fictitious device that it is, and to design instruction in which the principles of storing, extracting, converting, and transporting energy would be illustrated using features of the VST2000. Four of the designers were given the Operations problem first, followed by the Principles problem; the other four designers were given the Principles problem, followed by the Operations problem.

Analysis of Transcripts

Audio tapes of each participant's design activity were recorded and transcribed. Initial analysis involved dividing each transcript into episodes judged to involve a coherent segment of work related to a goal. Typically, there were about three episodes per page of the transcribed protocols.

We developed a coding scheme for characterizing the protocols. The scheme was developed iteratively as transcripts were coded: categories were added to distinguish between protocol segments that differed in theoretically meaningful ways; and categories were combined when they were difficult to distinguish. The final categories are listed in Table 1.

Table 1

Categories Used in Coding Protocols

Subproblems	Knowledge Types	Operators
Determine content	VST2000	Propose
Determine sequence	Science	Modify
Determine timing	Teach, general	Remove
Determine instructional transaction	Learn, general	Include information
Determine instructional resource	Available resources	Recap/reflect/evaluate/ monitor/justify
Determine constraints or requirements of the design task	Students	
Determine own constraints	Learn this	
Monitor progress or set out plan	Explicit teach this	
Clarify subject matter	Implicit teach this	
Content-relevant non-design issues	Instructional design task	
	Personal characteristics	

The attributes labelled subproblems, knowledge types, and operators are instances of the general framework described earlier. Subproblems consist of the component tasks that designers worked on while constructing their designs. The general character of the subproblems can be specified by considering the nature of the task. We formulated a tentative list and modified it in the course of our analysis. Five of the categories refer to constructing the design itself: (a) determining items of content to include in the instruction, (b) determining the sequence in which topics will be presented, (c) determining the amounts of time to be spent to different topics or activities, (d) determining the kinds of transactions to be used in presenting different materials--for example, the teacher can present an explanation or students can be given an exercise to carry out, and (e) determining instructional resources--for example, deciding to use the computer simulation for an exercise or deciding to show students the locations of the

energy sources in the actual vehicle. Four of the subproblem categories involve tasks that shape the problem space or are prerequisite to forming the design. These are (f) determining constraints or requirements of the task that has been presented--for example, clarifying the level of specificity that is needed for the design, (g) setting constraints on the design, (h) monitoring progress on the task or setting out a plan for the design process, and (i) clarifying the subject matter to be taught--for example, going back to the tutorial to understand better how one of the components of the VST2000 works. The final subproblem category includes nondesign activities related to the instruction, such as remembering personal experiences in learning about the VST2000, or teaching content similar to that involved in the design.

Operators are actions that change the design state or the problem space. The three state-transforming operators that we identified are: (a) proposing a feature of the design, (b) modifying a feature, and (c) removing a feature that had been proposed. Operators that involve shaping or navigating in the problem space are (d) adding information to the problem space, and (e) a collection of metadesign activities that we call recap/reflect/evaluate/monitor/justify (RREMJ).

The characteristics that we identified as knowledge types involve inferences about the kinds of information that the designers brought to bear in working on the instructional design. The knowledge types fit into five general types: subject-matter content, pedagogy, pedagogical content, task, and personal knowledge. The specific content types are (a) knowledge about the VST2000 and (b) knowledge about science and other relevant topics. Specific pedagogical types are (c) general knowledge or belief about teaching, (d) general knowledge or belief about learning, (e) knowledge about available resources for instruction, and (f) knowledge or belief about student characteristics. Pedagogical content knowledge involves information about teaching or learning the specific material of the instruction: (g) information about learning this material, (h) explicit statements regarding philosophy or beliefs about teaching this material, and (i) information about teaching this material in which the pedagogical content knowledge is implied rather than being explicitly stated. (j) Task knowledge includes references to the instructional design task that was presented to the designer by the interviewer. Finally, (k) the personal knowledge category includes information about oneself as a teacher or instructional designer, and about personal experiences.

Assignment of operator categories to segments of the transcripts was mutually exclusive. Assignment of subproblem and knowledge-type categories was not--a particular segment could be coded with more than one subproblem category or more than one knowledge-type category. One subproblem category was assigned as the dominant subproblem for each episode, but other subproblem categories could be assigned to segments within the episode as well.

Results

The coding scheme provides a set of categories that we were able to apply to the data with reasonable success. The subproblem and operator categories were relatively easy to apply with little ambiguity. The categories of knowledge types were more problematic, not surprisingly, because they involve a greater degree of inference. Even so, the three-attribute characterization of subproblems, operators, and knowledge types seems to do a reasonable job of characterizing our data.

We present only a sample of results in this report. A more detailed report (Greeno, Korpi, Jackson, & Michalchik, 1990) presents comparisons between the two design tasks, between performance on the first and second task done by individual designers, between

performance in different stages of the design process, and between designers who had completed the STEP program and those who were just beginning the program.

Figure 1 shows the lengths of protocol segments coded for each of the subproblem categories, summed across all the designers and tasks. The designers spent most of their activity determining the content of instruction and determining the nature of instructional transactions. Collectively, the five subproblems concerned with the design itself (the first five categories in Figure 1) occupied these designers more than the four categories of subproblems concerned with the nature of the problem space (the sixth through ninth categories), although the attention given to those subproblems was not insignificant.

Figure 2 shows the distribution of segment lengths across the categories of operators. The operator that occupied the designers the most was that of proposing material, which involves the various ways that designers introduced material into the design. A considerable amount of activity also was categorized according to the metadesign category of recapping, reflecting, evaluating, monitoring, and justifying. Some time was also spent putting new information into the problem space (shown as the I.I. category). The predominance of Propose and RREMJ reflect the fact that these designers spent almost all of their activity putting information into the design and reflecting upon information that they had put there. There was very little reversal in the activity of these designers, as indicated by the low values of the Modify and Remove categories.

Figure 3 presents the distribution of segment lengths for the categories of knowledge types, across designers and tasks. The most frequently coded category was "Teach This, Implicit," which included segments that were inferred to rely on knowledge of how this particular subject matter should be taught. This predominant category is in the general category of pedagogical content knowledge, rather than in pedagogical or in content knowledge, according to our definitions. There was a relatively small amount of activity that we inferred to rely on the general pedagogical category, according to our definitions, including general knowledge about teaching or learning, instructional resources, and the characteristics of students.

We constructed time-series graphs for each individual designer's activities, plotting the categories of subproblems, operators, and knowledge types across episodes.

A prototypical pattern of subproblems was observed in most of the designers. At the beginning of the problem, the designers spent a short time clarifying the task that the interviewer presented. The main part of the protocols was characterized by a strong emphasis on Determine Content, with small segments of Determine Sequence. The designers sometimes specified some Instructional Transaction or, less often, Instructional Resource, while proposing the content. Also, the designers sometimes commented on their progress on the design, as indicated by the presence of the Monitor subproblem. Late in the main design task, the designers often switched to Instructional Transaction and sometimes Instructional Resource, as the dominant subproblem, but continued to propose content and sequence. Toward the end of the transcripts, when participants were asked to review a structured sequence of topics, the designers usually followed a pattern that was consistent with this sequence.

The most salient distinctions among the designers in the way operators were used across the course of the design relate to the order in which the designers used the Propose and the RREMJ operators. Depending on the task, some of the designers used the Propose operator primarily and almost exclusively in the main section of the design task; in the review section they used RREMJ. In another frequent pattern, designers maintained nearly exclusive

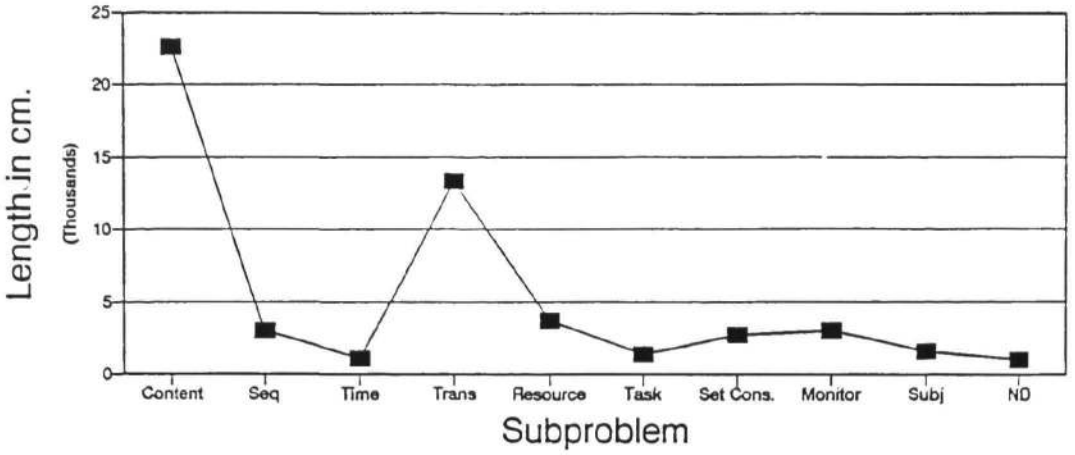


Figure 1. Subproblem lengths across tasks and designers.

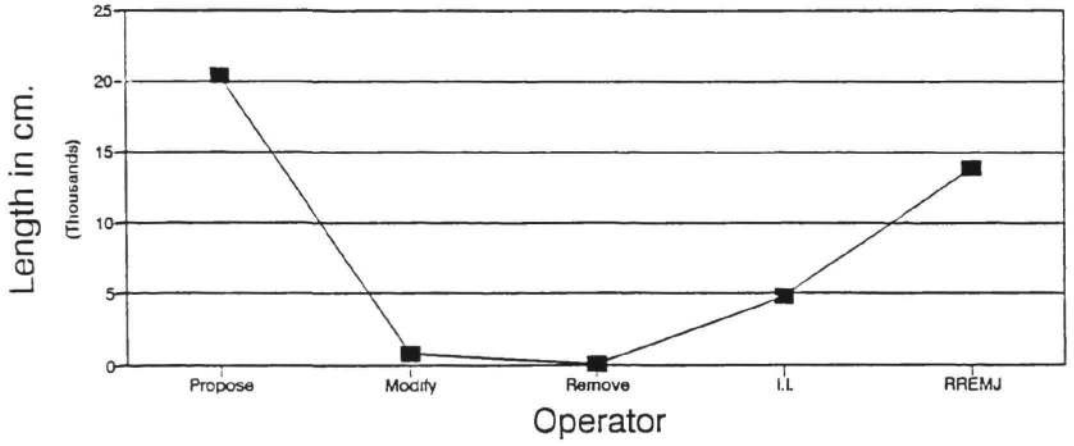


Figure 2. Operator lengths across tasks and designers.

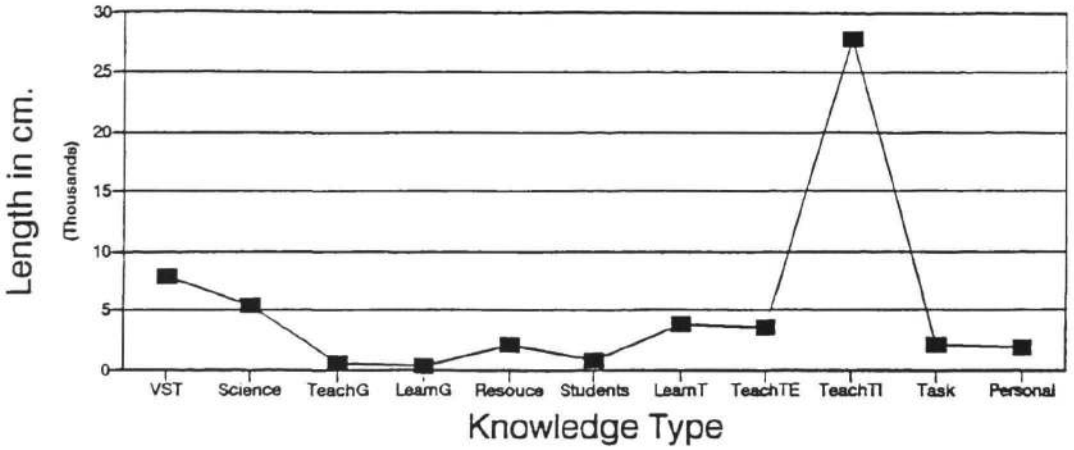


Figure 3. Knowledge type lengths across tasks and designers.

use of RREMJ in the review, and also used this operator one-third to one-half of the time in the main task.

The typical pattern of knowledge-type categories over time reflected the fact that very little use of purely pedagogical knowledge was judged to have occurred. In most of the protocols, therefore, the pattern of knowledge types across episodes showed concentrations of use of content knowledge and pedagogical content knowledge across all the episodes.

Discussion

Solution of a design problem can be characterized as finding some materials, and an arrangement of the materials, that satisfies some functional requirements. The form of our analysis, and many features of our characterization, parallel Ullman, Dietterich and Stauffer's (1988) analysis of problem solving in mechanical engineering design. In the performance of our designers, most of the activity involved determining the materials. The materials for a piece of instruction include the content of information to be included and the transactions by which that content will be communicated to students; work on these two subproblems dominated the designers' activity. The constructive nature of the task was reflected in the fact that about one-half of the accumulated transcript segments involved applying the operator of proposing material for the design. There was a considerable amount of metadesign activity, including reflection and evaluation of previously proposed material, but material was only rarely modified or removed once it had been placed into the design. The main type of knowledge used in design, according to our classification, was pedagogical content knowledge involving the designers' information and opinions about what knowledge needed to be taught and the importance or difficulty of that subject-matter in the instructional task.

Our analysis is compatible with other analyses of design problem solving that have been presented recently. Our distinctions between subproblems involving content and subproblems involving instructional sequences and transactions, and between knowledge about content and knowledge about pedagogy, are analogous to Bereiter and Scardamalia's (1987) distinction between activity in a content and in a rhetorical problem space in composing written essays. This distinction is also analogous to Kant's (1985) distinction between working in the problem space of a domain and in the problem space of algorithms in algorithm design. Simon (1973) also characterized separate problem spaces in his discussion of ill-structured problems. We prefer not to attach the label of different problem spaces to these distinctions, primarily because, in our data, work on the subproblems and use of the knowledge types appeared to be closely intertwined, rather than occurring in separable units of activity.

Acknowledgements

This research was supported by the Office of Naval Research (Contract N00014-88-K-0152). The authors thank Eamonn Kelly, Liza Loop, and Jim Winkles for assisting in the data collection, and Rena Kramer and Pam Weston for assisting with the coding.

References

Bereiter, C. & Scardamalia, M. (1987). The psychology of written composition. Hillsdale, NJ: Lawrence Earlbaum Associates.

Goel, V. & Pirolli, P. (1989). Motivating the notion of generic design within information processing theory: The design problem space. Unpublished manuscript, University of California, Berkeley, CA.

Greeno, J. G. & Berger, D. (1987). A model of functional knowledge and insight (Tech. Rep. No. GK-1). Berkeley, CA: University of California, School of Education.

Greeno, J. G. & Berger, D. (1990). Knowledge about system components and functions in diagnostic problem solving (Tech. Report). Stanford, CA: Stanford University, School of Education.

Greeno, J. G., Korpi, M. K., Jackson, D. N. & Michalchik, V. S. (1990). Processes and knowledge in designing instruction (Tech. Report). Stanford, CA: Stanford University, School of Education.

Greeno, J. G., Magone, M. E., & Chaiklin, S. (1979). Theory of constructions and set in problem solving. Memory andCognition, 7, 445-461.

Greeno, J. G. & Simon, H. A. (1988). Problem solving and reasoning. In R. C. Atkinson, R. J. Herrnstein, G. Lindzey, & R. D. Luce (Eds.), Steven's handbook of experimental psychology: Vol. 2: Learning and cognition (2nd ed.) (pp. 589-672). New York: Wiley & Sons.

Guindon, R. (1988). Software design tasks as ill-structured problems, software design as an opportunistic process (Report No. STP-214-88). Austin, TX: Microelectronics and Computer Technology Corporation.

Kant, E. (1985). Understanding and automating algorithm design. Proceedings of the Ninth International Joint Conferences on Artificial Intelligence (Vol. 2, pp. 1243-1253).

Newell, A. & Simon, H. A. (1972). Human problem solving. Englewood Cliffs, NJ: Prentice-Hall.

Reitman, W. R. (1965). Cognition and thought: An information-processing approach. New York: Wiley.

Simon, H. A. (1973). The structure of ill-structured problems. Artificial Intelligence, 4, 181-201.

Steler, D. S. & Newell, A. (1988). Integrating multiple sources into Designer-Soar, an automatic algorithm designer.Proceedings of the AAAI-88 Seventh National Conference on Artificial Intelligence (Vol. 1, pp. 8-13).

Ullman, D. G., Dietterich, T. G. & Stauffer, L. A. (1988). A model of the mechanical design process based on empirical data. Manuscript submitted for publication.

Psychological Test of an Algorithm for Recognizing Subjectivity in Narrative Text[1]

Gail A. Bruder
Department of Psychology
State University of New York at Buffalo
Buffalo, NY 14260
psybrude@ubvmsc.cc.buffalo.edu

Janyce M. Wiebe
Department of Computer Science
University of Toronto
Toronto, Canada M5S 1A4
wiebe@ai.toronto.edu

Abstract

This paper reports on a psychological study that tested part of an algorithm for recognizing subjectivity in third-person fictional narrative text (Wiebe 1990). In particular, we tested the part of the algorithm concerned with interpretations of *private-state sentences*, sentences about psychological states such as wanting and perceptual states such as seeing, by manipulating paragraph breaks and subjective elements. Six passages from natural narratives were selected. Two versions of each passage were used, the original version and an experimental one, which was created by moving paragraph breaks or removing subjective elements. Test statements were used to determine whether readers interpreted private-state sentences as private-state reports or as represented thoughts or perceptions. As predicted by the algorithm, deletion of subjective elements or discontinuities introduced by paragraph breaks decreased readers' tendencies to interpret private-state sentences as represented thoughts or perceptions.

1. Introduction.

Mental model or situation model approaches to narrative (cf. Johnson-Laird 1983; Bower & Morrow 1990) suggest that narrative comprehension includes the construction and use of a mental representation of the story or situation. As part of such an approach, we have suggested that readers construct and update a representation of the current spatio-temporal location of the story as it unfolds (Bruder et al. 1986; Rapaport et al. 1989). We call this spatio-temporal representation the "Deictic Center" and include in it a representation of the psychological point of view from which the events are portrayed (Uspensky 1973).

Our approach to the psychological point of view is based on Ann Banfield's (1982) theory of narrative sentences. Banfield characterizes narrative sentences as *objective* or *subjective*. Objective sentences objectively narrate events. Subjective sentences present the consciousness of an experiencing character within the story (called the *subjective character*). Two kinds are *represented thought* and *represented perception*, which portray characters' thoughts and perceptions, respectively (Banfield 1982). As Wiebe (1990) argues, sentences that report a character's *private state* such as wanting or seeing—that is, some perceptual, psychological, or experiential state typically not open to objective observation or verification—can also be considered to be subjective.

The following passage illustrates different kinds of subjective sentences.

(1)

[1.1]"What are you doing in here?" [1.2]*Suddenly she [Zoe] was furious with him [Joe].*

[1.3]"Spying, of course."

[1.4]"Well of all dumb things! [1.5]I thought you ran away." [1.6]*Joe Bunch was awful.* [Oneal, *War Work*, p. 130]

[1] This research was supported under NSF Grant No. IRI-8610517.

Sentence (1.2) is a private-state report: It reports Zoe's private state of being furious with Joe. Sentence (1.6) is a represented thought: It presents Zoe's thought and it expresses her evaluation of Joe (that he is awful). Represented perceptions appear in the following passage:

(2)

[2.1]Certainly, Dennys thought, anything would be better than this horrible-smelling place full of horrible little people.

[2.2]There was a brief whiff of fresh air. [2.3]A glimpse of a night sky crusted with stars. [L'Engle, *Many Waters*, p. 25]

Sentence (2.1) is Dennys's represented thought and (2.2) and (2.3) are Dennys's represented perceptions.

Wiebe (1990) presents an algorithm for recognizing subjective sentences and identifying their subjective characters, which is based on regularities in ways that texts initiate, continue, and resume a character's psychological point of view. In one part of this work, Wiebe argues that two interpretations of sentences that denote private states (i.e., *private-state sentences*) must be distinguished to track the psychological point of view: a private-state report and a represented thought or perception. The subjective character of a private-state report is the experiencer of the private state; the assertion that he or she experiences the state is assumed to be fictionally true (i.e., true in the fictional world). The subjective character of a represented thought or perception is the thinking or perceiving character; the sentence reflects his or her beliefs, whether or not it is a private-state sentence. These beliefs may or may not be fictionally true.

The interpretation of a private-state sentence may affect the identity of its subjective character. The subjective character of a private state *report* can always be identified from the sentence alone—he or she is the experiencer. In cases where the experiencer is not the last subjective character, the sentence initiates a new point of view. In contrast, the subjective character of a represented thought or perception typically has to be identified from the previous context.[2] Often, he or she is the subjective character of the last subjective sentence in the text, so the sentence continues or resumes that character's psychological point of view (an example is sentence (1.6)).[3]

Consider this passage:

(3)

[3.1]"Drown me?" Augustus said. [3.2]"Why if anybody had tried it, those girls would have clawed them to shreds." [3.3]He knew Call was mad, [3.4]but wasn't much inclined to humor him. [3.5]It was his dinner table as much as Call's, [3.6]and if Call didn't like the conversation he could go to bed.

[3.7]Call knew there was no point in arguing. [3.8]That was what Augustus wanted: argument. [3.9]He didn't really care what the question was, [3.10]and it made no great difference to him which side he was on. [3.11]He just plain loved to argue. [McMurtry, *Lonesome Dove*, p. 16]

Sentences (3.3)-(3.6) are Augustus's subjective sentences, and (3.7)-(3.11) are Call's.[4] Most of the private-state sentences in this passage have an experiencer who is not the last subjective character: The experiencer of (3.6) and (3.7) is Call, but the last subjective character when each is encountered is Augustus; and the experiencer of (3.8)-(3.11) is Augustus, but the last subjective character when each is encountered is Call. Only one is a private-state report, initiating a new point of view: sentence (3.7). The others are represented thoughts that continue the last subjective character's

[2] An exception is a represented thought with a narrative parenthetical, for example "Dennys thought" in (2.1).

[3] As the algorithm processes a text, it keeps track of *expected subjective characters*, characters who are likely to become subjective characters of later subjective sentences. When the algorithm identifies the subjective character from the previous context, it chooses an expected one. In the kinds of situations of concern in this paper, the last subjective character is always expected, and he or she is the only character who is. However, there are situations in which these conditions do not hold (see Wiebe 1990 and Wiebe forthcoming for details).

[4] Numbering within this passage reflects the actual units of input given to the algorithm, which are not always sentences. For example, the conjuncts of compound sentences are separate input units. By "sentence" we actually mean "input unit"

psychological point of view.

Wiebe (1990) identifies two factors that can affect the interpretation of private-state sentences. The first is the absence or presence of a paragraph break. The psychological point of view does not typically shift from one character to another without a paragraph break; thus, the lack of a paragraph break suggests that a shift has not occurred. In the case of a private-state sentence following a subjective sentence, where the experiencer is different from the last subjective character, the lack of a paragraph break separating them suggests that the private-state sentence is *not* a private-state report; if it were, then a shift would occur without a paragraph break. In this situation (that is, a private-state sentence that immediately follows a subjective sentence without a paragraph break), the algorithm identifies the subjective character to be the last subjective character, not the experiencer.

The second factor identified by Wiebe (1990) that can affect the interpretation of a private-state sentence is the appearance of a *subjective element* in the sentence.

There are many linguistic elements that can indicate that a sentence is subjective. Examples are evidentials such as 'evidently', evaluative adjectives such as 'silly' and 'foolish', and non-inverted questions. Although some of these elements always indicate that a sentence is subjective, others only potentially do so. A *subjective element*[5] is one that actually indicates that a sentence is subjective, in the context of use (see Wiebe 1990 for a description of how the algorithm decides if an element is subjective).

In a private-state sentence, a subjective element that does not appear within the scope of the private-state term (i.e., is *non-subordinated*) suggests that the sentence is a represented thought or perception, rather than a private-state report.

Ours is not the first psychological test of narrative point of view. In an early seminal study with human readers, Black, Turner, & Bower (1979) used deictic verbs in test sentences to create a point of view. They found that consistency in point of view between the subject of the first two sentences and the test sentence facilitated comprehension. Other more recent psychological research has also indicated the importance of point of view of the protagonist in narrative understanding (cf. Morrow, 1985; Morrow, Greenspan & Bower 1987; Bruder 1988). However, of these studies only Bruder used naturally occurring narrative, and her primary focus was on the spatial location signaled by deictic verbs.

This paper presents the results of an experiment with human readers which tested part of Wiebe's algorithm for recognizing subjective sentences and their subjective characters. Using natural narrative, we investigated the effect of paragraph breaks and non-subordinated subjective elements on the interpretation of private-state sentences. Given the redundancy of natural language, numerous factors may influence readers' interpretations, and human readers are neither as predictable nor as consistent as any particular algorithm. We therefore did not assume that manipulation of one such factor—i.e., paragraph breaks or a single subjective element—would change sentences from, for example, represented thoughts or perceptions to private-state reports for all subjects for all passages. However, we did predict that manipulations of such factors would influence the degree to which readers would interpret private-state sentences as represented thoughts or perceptions of another character.

We took passages from among those cited by Wiebe 1990 that contained private-state sentences. While these passages contained both private-state reports and represented thoughts or perceptions, most of the experimental manipulations were applied to represented thoughts or perceptions and were predicted to result in movement toward interpretation as private-state reports. We introduced paragraph breaks before private-state sentences and we removed subjective elements from private-state sentences. Each of the manipulations involved a private-state sentence that, as interpreted by the algorithm presented by Wiebe 1990, has an experiencer who is not the last subjective character. In some passages, the paragraph break that was inserted before the private-state sentence was moved

[5] This term is borrowed, but redefined, from Banfield (1982).

from an earlier point in the text in order to avoid having two paragraph breaks with only a single sentence between them. In these passages, the earlier sentence (i.e., the sentence from before which the paragraph break was moved) was also a private-state sentence; this allowed us to test the effect of removing a paragraph break on the interpretation of a private-state sentence.

Our predictions were as follows: Introduction of a paragraph break immediately before a private-state sentence would decrease the tendency for subjects to treat it as a represented thought or perception; removal of a subjective element would decrease the tendency for subjects to treat the private-state sentence as a represented thought or perception; and removal of a paragraph break from immediately before a private state sentence would decrease the tendency for subjects to select the experiencer as the subjective character of the sentence.

2. Method.

Subjects were 50 introductory psychology students participating for class credit. Because of missing responses on some of the test statements, the data of 6 subjects are not included in the analyses.

Materials were 6 passages from various sources including *Lonesome Dove*, by Larry McMurtry, *No One Hears But Him*, by Taylor Caldwell, and *The Magic of the Glits* by Carole Adler. There were two versions of each passage: In addition to the original version, there was an experimental one created by one of the manipulations described above. For half of the subjects, passages 1,4, and 5 were original and 2,3, and 6 were experimental; for the other half of the subjects, passages 1,4, and 5 were experimental and 2,3, and 6 were original.

For each passage, 4 or 5 test statements were written that referred to events in the passage. A few of them were "fillers", so that we would not always be asking about private states. The critical test statements were always in pairs, so that the interpretations of the targeted private-state sentence could be compared (the comparison is described in Section 3, below). There were a total of 10 critical test pairs for the 6 passages. Whether there were 1 or 2 test pairs per passage depended on the amount of text following the manipulation and whether we were testing two private-state sentences (as was the case when paragraph breaks were moved). In the two passages with the moved paragraph break, the break shifted from in front of one private-state to between it and the next private-state sentence. In these passages, we tested both private-state sentences. In the two passages with the newly introduced paragraph break, we had two test pairs referring to private states mentioned after the break. In the two passages with the private state sentence including a subjective element, we had a single test pair referring to the private-state mentioned in that sentence.

With two exceptions, all of the manipulations (introducing paragraph breaks and removing subjective elements) were predicted to shift subjects' interpretations from represented thoughts or perceptions to private-state reports. The exceptions occurred when we moved a paragraph break, thereby removing it from in front of one of the private-state sentences. In these cases, we predicted that the subjects would shift interpretation in the direction of a represented thought.[6] Again, we emphasize that we did not expect subjects to clearly change from complete agreement on one interpretation to complete agreement on the other; we were looking for predicted shifts in the degree to which they preferred one interpretation over the other.

Subjects were run in groups of about 10 and were allowed to take as much time as they wished. The instructions emphasized that there were no right or wrong answers, and that we were interested in their impression or understanding of the passages. On each page of the test booklet was one passage followed by the test statements. Subjects were to circle a number from 1 (strongly disagree) to 6 (strongly agree) to indicate whether they agreed with the test statement.

[6] In one, the sentence interpretation should, according to the algorithm, shift from a private-state report to a represented thought; in the other, the sentence should be interpreted as a represented thought in both the original and experimental versions, but the manipulation was expected to strengthen the represented thought interpretation.

3. Results and Discussion.

We determined whether the sentence was treated as a private-state report or a represented thought or perception by comparing the reader's agreement with each of two statements about the private state. One statement asserted that the private-state description is accurate. The other statement asserted that the private-state description is another character's belief or opinion. If a reader interprets a private-state sentence as a private-state report, then he or she should be more in agreement with the statement that the private state is true. If a reader interprets a private-state sentence as a represented thought or perception, then he or she should be more in agreement with the statement that another character believes that the private state is true. Our basic data was the difference in agreement values assigned to each of the pair of test statements that refer to a given private-state sentence in the passage.

Sample passages, test sentences, and mean agreement scores are presented in Table 1. For the analyses, we obtained the difference between the scores given to each member of a test pair. In order to facilitate the analysis we calculated the difference scores in a direction which, according to our predictions, would yield a larger value for those subjects in condition 2.

An analysis of variance for repeated measures (MANOVA) was performed on the difference scores from the ten test pairs. The difference between the two conditions was significant, $F(1,42)=4.21$, $p<.05$. There was also a large difference among passages, $F(9,378) = 28.32$, $p<.001$. As is evident in Table 1, this reflects differences across passages in the size of the difference between test statements. Because there was no interaction between passages and the conditions, $F<1$, this disparity does not weaken the conclusion that our manipulations influenced sentence interpretations; it merely confirms our expectation that many aspects of passages would influence our readers.

We performed an analysis of variance to test the significance of our manipulations across materials as well as subjects. With materials as a random factor, the manipulation was still significant, $F(1,9)=6.66$, $p<.03$.

We also looked at the interpretation of the original version of the passages. If, for a given subject, there was no difference between a test pair in agreement, then we could not determine which interpretation that particular subject obtained. Counting just those subjects who showed a difference in the test pair, we found that in 8 out of the 10 test pairs, more subjects interpreted the private-state sentence as predicted by the algorithm than did not. The other two test pairs were from passages testing subjective elements. Over all test pair comparisons, about half indicated that the original version was obtained, one quarter indicated that the other interpretation was obtained, and, for one quarter, there was no difference between the test pairs.

These data provide support for the algorithm, in particular in passages with paragraph break manipulations.[7] Interpretations of the original version of private-state sentences tended to be consistent with the algorithm and our manipulations influenced these interpretations in a direction predicted by the algorithm. Because there were fewer subjective element passages and one had a typographical error in one of the test statements, we are less confident in the adequacy of our tests of these passages.[8]

In future research, we plan to test the effects of subjective elements on the interpretation of private-state sentences. This will include both adding and deleting subjective elements. We also plan additional tests of manipulating paragraph breaks, including both adding and deleting such breaks

[7]The MANOVA analysis recalculated with just the difference scores for the 8 test pairs from the paragraph break passages still indicates a significant difference between versions in the direction predicted by the algorithm.

[8] The typographical error, using the plural rather than the singular for the sentence subject, may have decreased the likelihood readers would agree with that sentence. Since that test sentence was the represented thought interpretation, this may explain the failure to find the interpretation predicted by the algorithm in this passage. Because the error occurred in both versions, this passage is included in the analysis of difference scores: Any differences found could not be attributed to the typographical error, since it occurred in both versions.

around private-state sentences.

Acknowledgments. We would like to thank the members of the SUNY Buffalo Graduate Group in Cognitive Science and the SNePS Research Group for many discussions and ideas.

References

(1) Adler, Carole S. (1987), *The Magic of the Glits* (New York: Macmillan).

(2) Banfield, A. (1982), *Unspeakable Sentences: Narration and Representation in the Language of Fiction* (Boston: Routledge & Kegan Paul).

(3) Black, J.B., Turner, T. J., & Bower, G. H. (1979), "Point of View in Narrative Comprehension, Memory, and Production," *Journal of Verbal Learning and Verbal Behavior,* 18: 187-198.

(4) Bower, G.H. & Morrow, D.G. (1990), "Mental Models in Narrative Comprehension," *Science,* 247: 44-48.

(5) Bruder, G.A., Duchan, J.F., Rapaport, W.J., Segal, E.M., Shapiro, S.C., & Zubin, D.A. (1986), "Deictic Centers in Narrative: An Interdisciplinary Cognitive-Science Project," *Technical Report 86-20* (Buffalo: SUNY Buffalo Department of Computer Science).

(6) Bruder, G.A. (1988), "The Deictic Center and Sentence Interpretation in Natural Narrative," Paper presented at the *Psychonomic Society Meeting,* Chicago.

(7) Caldwell, Taylor (1966), *No One Hears But Him* (Garden City, NY: Doubleday).

(8) Johnson-Laird, P.N. (1983), *Mental Models* (Cambridge, MA: Harvard Univ. Press).

(9) L'Engle, Madeleine (1986), *Many Waters* (New York: Dell Publishing).

(10) McMurtry, Larry (1985), *Lonesome Dove* (New York: Simon & Schuster).

(11) Morrow, D.G. (1985), "Prepositions and Verb Aspect in Narrative Understanding," *Journal of Memory and Language,* 24: 390-404.

(12) Morrow, D.G., Greenspan, S.L., & Bower, G.H.(1987), "Accessibility and Situation Models in Narrative Comprehension," *Journal of Memory and Language,* 26: 165-187.

(13) Oneal, Zibby (1971), *War Work* (New York: Viking Press).

(14) Wiebe, J.M. (1990), "Recognizing Subjective Sentences: a Computational Investigation of Narrative Text," Ph.D. dissertation, Buffalo: SUNY Buffalo Department of Computer Science.

(15) Wiebe, J.M. (forthcoming), "Identifying Subjective Characters in Narrative," to appear in *Proceedings of the 13th International Conference on Computational Linguistics (COLING-88, Helsinki)* (Morristown, NJ: Assoc. for Computational Linguistics).

Table 1

Passages, Test Statements, and Mean Agreement Scores

"Drown me?" Augustus said. "Why if anybody had tried it, those girls would have clawed them to shreds." He knew Call was mad, but wasn't much inclined to humor him. It was his dinner table as much as Call's, and if Call didn't like the conversation he could go to bed. [*]

Call knew there was no point in arguing. That was what Augustus wanted: argument. He didn't really care what the question was, and it made no great difference to him which side he was on. He just plain loved to argue. [McMurtry, *Lonesome Dove*, p. 16]

[*experimental version moved paragraph break down one sentence to after "arguing".]

	Original	Experimental
Call felt that Augustus wanted an argument.	5.54	4.68
Augustus did want an argument.	4.82	4.54
It was Augustus opinion that Call knew there was no point in arguing.	3.14	3.18
Call did know that there was no point in arguing.	5.32	4.70

Well, he'd listen! And then freedom, like a kid again.

Dimly he heard the bell chime. But he was sunken in his anticipations. Then the girl said to him across the room in her sweet well-bred voice. "You are next." He started and looked up. They were alone. He winked at he impudently, showing his dimples. She went back to her reading. He yawned, stood up, pulled down his jacket, and sauntered to the door. He had an easy, boyish lope which he knew was very appealing to women. The girl was evidently [*] not impressed for she did not look up. [Caldwell, *No One Hears But Him*, p. 100]

[* experimental version removed "evidently"]

The boy felt that the girl was not impressed.	3.50	3.05
The girl was not impressed	5.18	5.41

In a panic, he shook her. "Lynette, it's me, Jeremy. Wake up now!"

At first she was rigid. Then she turned around and dug her sharp-boned little face into his shoulder. She shook with sobs. Jeremy didn't know how to stop them or even if he should try to. He just sat getting wet from her hot tears. Anxiously he waited. She felt fragile in his arms as if anything at all might break her.

The flames sank away. Only the glowing red treasure heap of embers was left and the cold mist at his back and the sense of being in a lang with no familiar landmarks in the dark. [*] How could she not know her mother was dead? She had to know it. [Adler, *Magic of the Glits*, p. 45]

[* experimental version added a paragraph break]

Jeremy wondered how Lynette could not have known her mother was dead.	5.18	4.82
Lynette wondered how she could not have known her mother was dead.	1.95	2.27
Jeremy felt Lynette had known about her mother's death subconsciously.	4.55	4.36
Lynette felt she had known about her mother's death subconsciously.	3.14	3.86

Cross-Domain Transfer of Planning Strategies: Alternative Approaches

Bruce Krulwich, Gregg Collins, and Lawrence Birnbaum
Northwestern University
The Institute for the Learning Sciences and
Department of Electrical Engineering and Computer Science
Evanston, Illinois

ABSTRACT

We discuss the problem of transferring learned knowledge across domains, and characterize two possible approaches. *Transfer through reoperationalization* involves learning concepts in a domain-specific form and transferring them to other domains by recharacterizing them in each domain as necessary. *Abstraction-based transfer* involves learning concepts at a high level of abstraction to facilitate transferring them to other domains without recharacterization. We discuss these approaches and present an example of the abstraction-based transfer of a method of *projection*, or selective lookahead, from the game of chess to the game of checkers, as implemented in our test-bed system for failure-driven learning in planning domains. We then discuss a continuum of abstraction to characterize learned concepts, and propose a corresponding continuum characterizing the time at which the computation necessary for cross-domain transfer is accomplished.

1 Introduction

Human beings have the ability to learn a concept in one domain and apply it in a different domain. Achieving such *cross-domain transfer* is a current goal of machine learning systems as well. Transferring a concept learned in one domain into a different domain involves both determining that the concept is applicable to the new domain, and casting it in a form that is appropriate to that domain.

Previous studies of the human ability to transfer knowledge between domains have investigated the way in which this ability depends on the learning method employed and on the learner's prior knowledge [Katona, 1940; Hilgard, Irvine, and Whipple, 1953; Mayer and Greeno, 1972]. In particular, it has been observed that the ability to transfer learned concepts between domains depends on the level of abstraction at which the concepts are represented and indexed [Hilgard, Ergren, and Irvine, 1954; Mayer and Greeno, 1972; Singley and Anderson, 1989]. Brown has proposed a hypothetical continuum of knowledge abstraction in the context of transfer that ranges from complete *theories*, explaining how a concept relates to the other knowledge of the system, to *arbitrary solutions* that do not include any explanation of either correctness or appropriateness, and includes a number of intermediate levels [Brown, 1989].

A computer system might represent and index concepts that it learns at any of these levels of knowledge abstraction, and its ability to transfer this learned knowledge to another domain would depend upon the level employed. If the system formulates a concept in terms of a general theory, the concept will, as a result, be applicable to any domain in which the

theory is applicable. If, on the other hand, the system develops only a partial explanation of a concept, without tying it to a more general theory, it can nevertheless transfer this concept to domains in which the explanation is applicable. Concepts for which no explanation is generated can of course be applied to other domains only on a hit-or-miss basis.

The approach that must be taken to transferring a concept across domains depends upon the level of abstraction of the vocabulary in which it is expressed. In particular, we can distinguish two basic approaches, each appropriate to a different end of the abstraction continuum. *Transfer through reoperationalization* applies when transferring a concept that has not been abstracted out of its particular domain, and which therefore must be recharacterized in terms appropriate to the new domain. *Abstraction-based transfer*, on the other hand, is appropriate when transferring a concept expressed in terms of a general theory that is already applicable to the new domain. In this paper we will discuss these two approaches, and the trade-off between them, and present a computer implementation of *abstraction-based transfer* in competitive planning.

2 Transfer through reoperationalization

In *transfer through reoperationalization*, a concept is initially learned in a vocabulary specific to a particular domain, and must therefore be recharacterized, or *reoperationalized* (see, e.g., [Mostow, 1983]), when it is needed in another domain. The distinction between this approach to transfer and others is that no processing is done to facilitate transfer of a concept before the concept is needed in the new domain. This approach has been taken by a number of researchers, include many in the areas of case-based reasoning (see, e.g., [Kolodner, 1988; Riesbeck and Schank, 1989]) and analogy (see, e.g., [Gentner, 1983; Carbonell, 1986]). In this approach, transfer is accomplished in two steps: first, determining that a particular concept learned in one domain is applicable to another domain, and second, expressing the concept in the vocabulary of the new domain.

This approach raises several difficult issues. The first step, determining that a concept is applicable, leads to the *indexing problem*, the problem of organizing domain-specific concepts in memory in a way that facilitates retrieval in appropriate situations in other domains (see, e.g., [Schank, 1982]). The second step, expressing the concept in the new domain, requires a theory of how to map concepts between the two domains (see, e.g., [Gentner, 1983]). While a number of fruitful approaches to these problems have been developed, the process of applying a concept in a new domain remains computationally expensive. On the other hand, the benefit of transfer by reoperationalization is that no effort is expended when the concept is learned to prepare it for use in domains other than the one in which the system is currently operating.

3 Abstraction-based transfer

Abstraction-based transfer involves learning new concepts in a form that facilitates their transfer directly to other relevant domains. This processing involves immediately generalizing the concept to the highest level of abstraction that is supported by its explanation (see, e.g., [DeJong and Mooney, 1986; Mitchell, Keller, and Kedar-Cabelli, 1986]). When the concept is applied in the new domain, it can either be used directly or specialized to the vocabulary of the new domain.

This approach to transfer has the advantage of minimizing the cost of indexing and mapping, since applicable concepts will already be expressed in terms relevant to the current domain. On the other hand, it raises the question of how, or even *whether*, the concept can be learned in terms that are abstract enough to apply in all domains for which the concept might be appropriate.

4 An implementation of abstraction-based transfer

Cross-domain transfer of knowledge is a central concern of our current research in failure-driven learning in planning domains (see, e.g., [Birnbaum and Collins, 1988]). To explore these issues more concretely, we have implemented a game-playing model for two player turn-taking games, along with specific rules for such games as chess and checkers, within our test-bed system for exploring failure-driven learning in planning domains (see, e.g., [Collins, Birnbaum, and Krulwich, 1989]). This system provides a unified framework for inference, justification maintenance, expectation monitoring, and explanation of failures. All of the beliefs, rules, and expectations in the system are tagged with justification structures indicating the basis for the system's belief in their correctness. The system learns from its failures [Sussman, 1975; Schank, 1982; Hayes-Roth, 1983; Kolodner, 1987; Hammond, 1989] by monitoring the truth of any predictions it makes about the world in the course of its decision-making. When any of these expectations fail, the system constructs an explanation of the failure using the justification for that expectation. Should this explanation reveal a bug in the system's decision-making mechanism, the system will attempt to modify its rules to correct the problem, thus avoiding similar failures in the future.

Decision-making in our system is accomplished by a fairly general, albeit rudimentary, mechanism. First, the decision component computes the opportunities available to the computer, using game-specific notions of threats and moves. The results of each possible move are then predicted by the projection component, which uses a general method for all games. These results are then ranked by the evaluation component, which is again specific to the game being played. The computer then chooses the highest-ranked such move.

We will focus here on the second of these three components, the *projection* component. Because the general method of projection used by our system is game-independent, it is capable of supporting abstraction-based transfer between games. In general, we would like our system to project the results of a move as far into the future as possible, because this will improve its value as an estimator of the quality of that move. However, the further into the future a projection is carried, the more expensive it will be to compute, and a system will have no basis for making an *a priori* decision about this tradeoff upon entering a new domain. Our approach to this problem is to have the system start out with a very simple projection mechanism that it will augment as necessary [Krulwich, Collins, and Birnbaum, 1989]. In particular, our system will initially project the results of a move by assuming that on the following turn the opponent will make the best move that is *currently* available to it. The rule that implements this projection method, which is shown in figure 1, says roughly that *the situation resulting from making a move is the situation after making the move and after the opponent makes the move which is the best move at the current time.* This unsophisticated procedure considerably simplifies the projection problem, because it obviates the need to repeatedly recompute the opponent's response for each move contemplated by the planner. However, its validity, and hence the validity of the predictions that it generates concerning the opponent's moves, depends upon an

```
(def-brule proj-factor-2  <...variable declarations...>
      (project-factor proj-factor-1.5 world move player result 1)
   <=
      (and (= world (world-at-time time))
           (decide (player-opponent player)
                   (possible-moves-at-time (player-opponent player) time)
                   world simple-dec-factors opp-move)
           (= result (world-after-move opp-move (world-after-move move world))) ))
```

Figure 1: Initial Projection Method

assumption that nothing will occur to enable the opponent to make a higher-priority move than those currently available to it. This is an instance of what is sometimes referred to as a *persistence assumption,* namely, an assumption that things will stay as they are as much as possible. This assumption is itself justified by a conjunction that says, roughly, *Nothing will happen to give him a better move because, (1) he can't do anything to give himself one, (2) no outside forces will give him one, and (3) I won't do anything to give him one.* These assumptions make up the justification for the projection method in figure 1. These assumptions are clearly not always true; the problem for our system is to determine the situations in which they are not true, and hence in which a more sophisticated projection method is necessary.

Consider how our system would behave in the situation shown on the partial chess board in figure 2a. Taking the opponent's knight with the rook looks like the best move, because the computer expects that the opponent will take *its* knight in the following turn, and it believes that trades are to its benefit. However, the opponent will not make this move, as we can see in figure 2c, because the computer's move gives him the opportunity to take the computer's rook, which is a more valuable piece than the knight. When the opponent makes his move to take the computer's rook, the system's expectation about the opponent's move—that he will take the knight—will fail.

In response to this expectation failure, the system will analyze the justification for the faulty expectation in order to explain the error. Traversing this justification will lead it to test the assumption that *the computer won't do anything to give the opponent a better move,* and the system will find that this belief is to blame for the expectation failure. In particular, the computer's own move enabled the better move that the opponent made. The system responds to this by patching its projection rule to take into account the possibility that

```
++ -- OP -- ++        ++ -- OP -- ++        ++ -- ++ -- ++
-- ON -- CN --        -- CR -- CN --        -- OP -- CN --
++ -- ++ -- ++        ++ -- ++ -- ++        ++ -- ++ -- ++
-- ++ -- ++ --        -- ++ -- ++ --        -- ++ -- ++ --
++ CR ++ -- ++        ++ -- ++ -- ++        ++ -- ++ -- ++

 (a) RxN   ...         (b) ...   BxR         (c) ...   ...
```

Figure 2: Faulty Projection in Chess: Computer to Move at Start

```
(def-brule proj-factor-3  <...variable declarations...>
   (project-factor proj-factor-1.5-mod world move player result 1)
 <=
   (and (= world (world-at-time time))
        (decide (player-opponent player)
                (possible-moves-at-time (player-opponent player) time)
                world simple-dec-factors opp-move)
        (= result (world-after-move opp-move (world-after-move move world)))
        (no (and (move-enables-move move better-move)
                 (move-possible better-move (current-time))
                 (move-legal better-move)
                 (evaluate result eval-factors (player-opponent player) orig-value)
                 (evaluate (world-after-move better-move (world-after-move move world))
                           eval-factors (player-opponent player) better-value)
                 (> better-value orig-value)  )) ) )
```

Figure 3: The Modified Projection Method

the computer's move enables a better move for the opponent, and to ignore the persistance assumption in such cases. This improved projection method, shown in figure 3, is the old method with the added condition that the computer's move not enable a better move for the opponent. This modification to the projection method will prevent the computer from making the same mistake in the future. When placed in the same situation as before, it will instead move its threatened knight to safety (as shown in figure 4).

Transferring to another domain

While many aspects of the process of projection are specific to the game of chess, the system's description of the improved projection process is as general as the original projection method was, and is thus expressed in vocabulary that is applicable to all turn-taking games. This should enable the system to use *abstraction-based transfer* to apply the more sophisticated method to similar games. To test this, we disabled the learning component of the system, and put the system in an analogous situation in the game of checkers. Using the original projection method described above, the system played as shown in figure 5. The computer predicts that the opponent's move will be to jump the computer's piece in the upper right-hand corner. As a result, the computer decides that its own best move would

```
++ -- OP -- ++        ++ -- OP -- ++
-- ON -- CN --        -- ON -- ++ --
++ -- ++ -- ++        ++ -- ++ -- ++
-- ++ -- ++ --        -- ++ -- ++ CN
++ CR ++ -- ++        ++ CR ++ -- ++

      (a)                 (b)
```

Figure 4: Repaired Projection in Chess: Computer to Move at Start

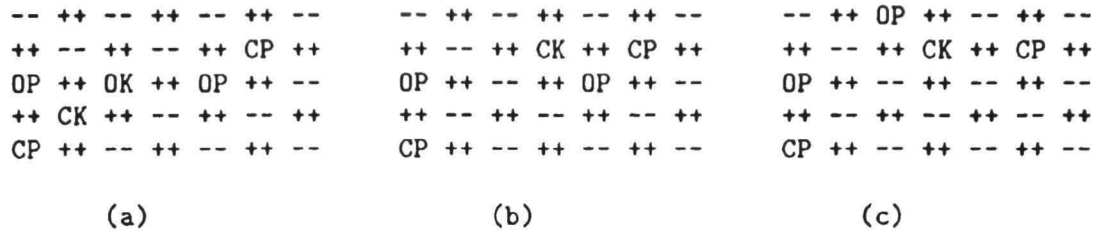


Figure 5: Faulty Projection in Checkers: Computer to Move at Start

be to take the opponent's king with its king. As we can see in figure 5b, the computer's prediction fails for the same reason that the chess prediction failed in the previous example namely that the computer's move enabled a better move for its opponent.

Next, we re-enabled the learning component of the system, and ran it through the chess situation described previously, allowing it to learn the improved projection rule described in figure 3. When we subsequently reran the checkers situation, the computer applied what it learned, and improved its behavior, as shown in figure 6.

5 Conclusion

Both approaches to transfer that we have discussed have their advantages and disadvantages. *Transfer by reoperationalization* requires realizing that the concept to be transferred is appropriate to the new domain, which involves computationally expensive processing at the time of problem solving in order to retrieve and apply the concept appropriately. On the other hand, *abstraction-based transfer* requires expressing the concept at a high enough level of abstraction to allow its utilization in all appropriate domains, which again involves computationally expensive processing, in this case at the time the concept is learned. It seems clear that the optimal approach will lie somewhere in between these two extremes.

Our research has, in part, been an exploration of this trade-off. One intermediate approach we have been pursuing is *explanatory transfer*, in which a concept is learned in response to a failure, and is generalized as much as is necessary to explain that failure. When similar failures occur in other domains, the concept may be retrieved and applied in the

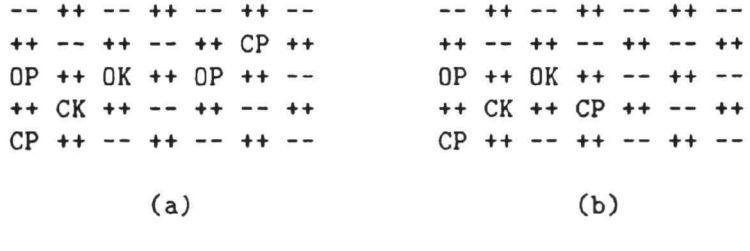


Figure 6: Repaired Projection in Checkers: Computer to Move at Start

course of constructing an explanation for the current failure [Collins and Birnbaum, 1988; Kass, 1989].

This approach combines several of the advantages of each of the previous two approaches. As in *abstraction-based transfer*, the concepts are generalized when they are learned, which reduces the computational effort necessary in retrieving and applying them in new domains. However, as in *transfer through reoperationalization*, the applicability of concepts to other domains need not be determined at the time the concept is learned. Furthermore, determining the applicability of the concepts to be transferred is more focussed—by the need to explain a particular failure in the new domain—than it is in the other approaches. Future research is necessary to develop other approaches to transfer that are between the two extremes, and to determine the areas in which the different approaches are applicable.

Acknowledgments: We would like to thank Matt Brand, Ann Holum Faillettaz, Mike Freed, Eric Jones, and Dick Osgood for many useful discussions. This work was supported in part by the Office of Naval Research under contract N00014-89-J-3217, and by the Defense Advanced Research Projects Agency, monitored by the Air Force Office of Scientific Research under contract F49620-88-C-0058.. The Institute for the Learning Sciences was established in 1989 with the support of Andersen Consulting, part of The Arthur Andersen Worldwide Organization.

6 References

Birnbaum, L., and Collins, G. 1988. The transfer of experience across planning domains through the acquisition of abstract strategies. [Kolodner, 1988], pp. 61-79.

Brown, A. L. 1989. Analogical learning and transfer: What develops? [Vosniadou and Ortony, 1989], chapter 14, pp. 369-412.

Carbonell, J. 1986. Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. *Machine Learning: An Artificial Intelligence Approach, Vol. 2*, Morgan Kaufmann, Los Altos, CA, pp. 371-392.

Collins, G., and Birnbaum, L. 1988. Learning strategic concepts in competitive planning: An explanation-based approach to the transfer of knowledge across domains. Research report no. UIUCDCS-R-88-1443, University of Illinois, Dept. of Computer Science, Urbana, IL, 1988.

Collins, G., Birnbaum, L., and Krulwich, B. 1989. An adaptive model of decision-making in planning. *Proceedings of the Eleventh IJCAI*, Detroit, MI, pp. 511-516.

DeJong, G., and Mooney, R. 1986. Explanation-based learning: An alternative view. *Machine Learning*, vol. 1, pp. 145-176

Gentner, D. 1983. Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, vol. 7, pp. 155-170.

Hammond, K. 1989. *Case-Based Planning: Viewing Planning as a Memory Task.* Academic Press, San Diego, CA.

Hayes-Roth, F. 1983. Using proofs and refutations to learn from experience. In R. Michalski, J. Carbonell, and T. Mitchell, eds., *Machine Learning: An Artificial Intelligence Approach, Vol. 1*, Tioga, Palo Alto, CA, pp. 221-240.

Hilgard, E., Ergren, R., and Irvine, R. 1954. Errors in transfer following learning by understanding: Further studies with Katona's card trick experiments. *Journal of Experimental Psychology*, vol. 47, pp. 457-464.

Hilgard, E., Irvine, R., and Whipple, J. 1953. Rote memorization, understanding, and transfer: An extension of Katona's card trick experiment. *Journal of Experimental Psychology*, vol. 46, pp. 288-292.

Kass, A. 1989. Adaptation-based explanation: Extending script/frame theory to handle novel input. *Proceedings of the Eleventh IJCAI*, Detroit, MI, pp. 141-147.

Katona, G. 1940. *Organizing and memorizing.* Columbia University Press, New York.

Kolodner, J. 1987. Capitalizing on failure through case-based inference. *Proceedings of the Nonth Cognitive Science Conference*, Seattle, WA, pp. 715-726.

Kolodner, J. 1988. *Proceedings of the 1988 Workshop on Case-Based Reasoning*, Morgan Kaufmann, San Mateo, CA.

Krulwich, B., Collins, G., and Birnbaum, L. 1989. Improving decision-making on the basis of experience. *Proceedings of the Sixth International Workshop on Machine Learning*, Ithaca, NY, pp. 55-57.

Mayer, R., and Greeno, J. 1972. Structural differences between learning outcomes produced by different instructional methods. *Journal of Educational Psychology*, vol. 63, pp. 165-173.

Mitchell, T., Keller, R., and Kedar-Cabelli, S. 1986. Explanation-based generalization: A unifying view. *Machine Learning*, vol. 1, pp. 47-80

Mostow, D. 1983. Machine transformation of advice into a heuristic search procedure. In R. Michalski, J. Carbonell, and T. Mitchell, eds., *Machine Learning: An Artificial Intelligence Approach, Vol. 1*, Tioga, Palo Alto, CA, pp. 367-403.

Riesbeck, C., and Schank, R. 1989. *Inside Case-Based Reasoning.* Lawrence Erlbaum Associates, Hillsdale, NJ.

Schank, R. 1982. *Dynamic Memory: A Theory of Reminding and Learning in Computers and People.* Cambridge University Press, Cambridge, England.

Simmons, R. 1988. A theory of debugging plans and interpretations. *Proceedings of the 1988 AAAI Conference*, St. Paul, MN, pp. 94-99.

Singley, K., and Anderson, J. 1989. *The Transfer of Cognitive Skill.* Harvard University Press, Cambridge, MA.

Sussman, G. 1975. *A Computer Model of Skill Acquisition.* American Elsevier, New York.

Vosniadou, S., and Ortony, A. 1989. *Similarity and Analogical Reasoning.* Cambridge University Press, Cambridge, England.

The Effect of Alternative Representations of Relational Structure on Analogical Access

Catherine A. Clement
Psychology Department, Eastern Kentucky University

Abstract

Retrieval of an appropriate analogy from memory is often difficult because the structure common to two analogous domains is embedded in specific contexts that differ at the surface level. The present study examines an aspect of domain representations that may affect the access of analogs in memory. Subjects were asked to identify analogies between new and previously learned passages. Passages varied in the manner in which analogous relations were described. In all passages the relations were embedded in a particular context that was dissimilar at the surface level between analogs. However, the expression of relations within a passage varied in level of abstraction. In "abstract" passages relations were lexicalized with relatively abstract terms and were described with little domain specific detail. In "specific" passages more specific terms were used and extensive domain specific detail was given about how relations were instantiated within the domain. In "mixed" passages both abstract and specific descriptions of relations were given. Subjects reading abstract passages were best at identifying analogies. The present results suggest that even though analogous relations are embedded in dissimilar contexts, the way in which those relations themselves are represented can affect analogical access. Subjects are relatively successful at analogical access when the relations are represented in a relatively general and sparse form.

The flexibility of the cognitive system is displayed when analogies are drawn between disparate domains, for example, when we transfer our knowledge about a hydraulic system to an electric circuit. But how do we make these connections? What leads us to notice that electricity and water flow can be compared?

Many investigators have found that recognizing analogical similarity between events, passages, or problems from disparate domains is often difficult. Subjects often fail to apply a previously learned task when presented with an analogous task (Reed, Ernst, & Banerji, 1974; Gick and Holyoak, 1980; 1983; Gentner and Landers, 1986; Ross, 1987). The difficulty seems to lie in the access process, that is, in spontaneously noticing the analogical similarity between a current task and a prior task represented in long term memory. Once subjects are presented with the two tasks for comparison, they have relatively little difficulty mapping their similarities.

The difficulty of analogical access is not surprising when we consider the features of an analogy between disparate domains. First, in an analogy the two domains being compared are not similar in terms of salient contextual or "surface features" such as concrete object attributes (Gentner, 1983). For example, electricity does not share "wetness" with water. Rather the two domains are only similar in terms of underlying structural relationships. Second, an analogy draws a correspondence between two specific domains rather than between an explicit, general principle or schema and the target domain. In an explicit principle, objects or arguments have become variables, and the relations are no longer embedded in a specific context (Gentner, 1983). Often we rely on analogies when such general principles or schemas are unavailable. A schema is induced only after an analogy has been discovered and used (Gentner, 1989).

Investigators have found that access of prior tasks represented in memory is strongly influenced by surface features shared with a target situation, (e.g. Ross, 1984, 1987; Lewis and Anderson, 1985; Gentner and Landers, 1986; Rattermann and Gentner, 1987; Holyoak and Koh, 1987). When only relational features are shared, prior tasks are often not easily retrieved. However, subjects can access relational structures when these structures are no longer embedded in specific contexts. Gick and Holyoak (1983) found that subjects could transfer a solution schema from base analogs to a target problem that shared few surface similarities with the analogs. However, transfer was found only when subjects had been given two base domains and had abstracted an explicit schema (characterizing the analogous structure) from the two domains. Chen

and Daehler (1989) and Clement (1987) found similar results. Thus access of explicit schemas, independent of the specific analogs, is easier than retrieval of specific cases in which the schemas are implicit. Bassok and Holyoak (1989) present a related finding that transfer of structures from mathematics to physics was easier than transfer of structures from physics to mathematics. Mathematics appears to allow essentially domain independent representations of structures, whereas in physics the structures are linked to particular contexts.

Thus, analogical access is difficult because the structure common to two analogous domains is embedded in specific contexts that differ at the surface level. Yet successful access of analogs can occur. Presumably when successful access occurs, the specific domains are represented in a manner that highlights the relevant relational similarities, and subjects are easily able to disregard surface dissimilarities (Novick, 1988; Faries & Reiser, 1989; Gick and Holyoak, 1983; Gentner,1989; Seifert, McKoon, Abelson, & Ratcliff, 1986). This does not imply that access requires the induction of an explicit domain-independent schema. Schema induction may be the last stage of analogical processing (Gentner, 1989). How can we account for access of analogs when subjects have not yet developed domain-independent schemas that characterize the analogous domains?

A key factor in analogical access may be the manner in which the analogous relations are represented within a domain. Even though analogous relations are linked to a particular context, their representation within that context can vary. For example, the relations may be represented in more or less domain specific format. Often analogous relations may only appear similar between domains when they are translated from relatively domain-specific terms to more general terms. For spontaneous access, it may be necessary that the analogous relations be already represented in a form that allows relatively direct detection of their similarity.

There are several ways in which the relations within analogous domains can be represented in a more or less compatible way. For example, if the relations are lexicalized with verbs, these verbs can be relatively specific or abstract. If they are specific, the general components must be differentiated from domain-specific components (such as the manner or instrument entailed by the verb) before cross-domain similarities can be detected. This may require viewing the verb as an instance of a more abstract concept, or decomposing the verb until a common core component is found (Gentner, 1989). As an example of the contrast between relatively abstract or domain-specific relational verbs, imagine a creature that "gathers" food. Either a general term like "gathers" can represent the relation, or a more specific term can be used that specifies how the gathering occurs, e.g. "sucks". If the creature is analogous to something else that gathers but does not suck, the more general term will allow easier detection of the analogy.

The representation of a relation may include a more or less detailed account of how the relation is instantiated within a domain. Again in order to recognize cross-domain similarities, these relations may have to be re-represented. They may either be lexicalized with a relatively abstract term, or a sparser, more general description may be substituted. This may require synthesizing detail to arrive at a higher level of description, or deleting detail to isolate a relevant general idea. For example, suppose a creature tests the food supply in his habitat. Detail about how this testing proceeds could be given, and it could be left for the reasoner to integrate the information and conceptualize it as "testing". In contrast, the relation could be lexicalized directly with the term "test".

In the present research the descriptions of relations in analogous domains was manipulated to make the similarity between domains more apparent. Domain descriptions were either abstract or specific. Although the basic content was the same, the abstract descriptions used more abstract words to convey key relations. Also, the abstract descriptions contained less domain-specific detail about how relations were instantiated. It should be made clear that the abstract descriptions were not domain-independent schemas. The relations were still linked to particular contexts. A third condition was also included in which both abstract and specific expressions of relations were included in a domain description.

This manipulation of the level of abstraction of analogous relations should affect how easily subjects can detect a structural similarity between analogous domains. If access processes rely on

the relatively direct detection of similarity, subjects with the abstract domain descriptions should be more likely to access a base domain in the presence of a target than subjects with specific domain descriptions.

Method

Design

Adult subjects learned base passages during a Learning Session. Two days later, during an Analogy Session, subjects had to match the base passages to target passages. We assessed subjects' ability to both access the correct base for a given target passage, and to work out the mapping between an identified analogous pair. Base and target passages differed for three groups: (a) For the Specific Group (n=28) all base and target passages received were "specific". (b) For the Abstract Group (n=28) all base and target passages received were "abstract". (c) For the Mixed Group (n=28) all base and target passages were "mixed" -- they included both abstract and specific descriptions of relations.

Materials

Analogies. Five analogies were developed each consisting of a base and target passage. Each passage was one paragraph in length and described the activity of a fictional creature or object. Table 1 outlines the common relations in an example analogy between two domains called "The Tams" and "The Satellites".

Table 1
Analogous Relations in *The Tams* and *The Satellites*

Gathering:
Tams: Creatures called Tams eat minerals from rocks.
Satellites: Satellites take photographs of planets.

Relocation:
Tams: When the minerals are gone from one place, the Tams move on.
Satellites: When an orbit is complete, (no more new photographs to be taken) the Satellite moves to a new path.

Testing Supplies:
Tams: At a new spot the Tam sees if minerals are plentiful.
Satellites: At a new path, the Satellite determines whether new pictures can be taken.

Staying or Moving on:
Tams: If the minerals are plentiful, the Tam stays and eats. Otherwise it moves on.
Satellites: If the pictures are fresh, the Satellite stays and takes pictures. Otherwise it moves on.

Tables 2 and 3 illustrate the actual wording used to describe some of the relations in the abstract and specific versions of the Tams and Satellites passages. Notice that more or less abstract verbs are used to describe relations, and that more or less detail is given about how relations are instantiated in the domain. The mixed version simply combined the content of the abstract and specific versions. For each analogy, which passage served as the base (given during the Learning Session), and which served as the target (given during the Analogy Session) was varied within each group.

Surface matches. Each passage was designed to share a salient surface similarity to one other passage to which it was not analogous. This surface match included a similarity in object type, object attribute and/or simple relation. For example, a passage not analogous to the Tams was similar to the Tams in that the central characters were also "fierce" and also lived among rocks. No other information in the two passages was similar. These surface matches were present in each condition-Abstract, Specific and Mixed.

Table 2
Examples of the Instantiation of Analogous Relations in the Abstract and Specific Versions of *The Tams.*

Gathering

Abstract *Tams are fierce little creatures that gather and consume a mineral called feldspar found in rocks. A Tam uses its underbelly to gather and bring in the feldspar.*

Specific *Tams are fierce little creatures that voraciously eat a mineral called feldspar found in rocks. A Tam has an abrasive underbelly that hurriedly wriggles back and forth to loosen minerals from a rock. The belly then slurps in loosened feldspar.*

Testing Supplies (after moving to a new location)

Abstract *As soon as the Tam reaches a new spot, it tests the available mineral supply.*

Specific *As soon as the Tam discovers a new spot, it sees how much feldspar can be loosened with a few quick wriggles of its underbelly.*

Table 3
Examples of the Instantiation of Analogous Relations in the Abstract and Specific Versions of The Satellites.

Gathering

Abstract *A special satellite is used to collect data about planets. This satellite collects pictures with highly sensitive cameras. The satellite follows a path around a planet taking pictures at frequent intervals.*

Specific *A special satellite is used to provide data about planets. The satellite slowly spins a path around a planet, snapping pictures at frequent intervals with highly sensitive cameras. Every few seconds, a camera eye opens and new film is exposed.*

Testing Supplies (after moving to a new location)

Abstract *Sometimes the second path is too close to the first path, and the pictures taken are duplicates. Therefore, whenever the satellite begins a new path, it evaluates whether or not it is collecting fresh pictures.*

Specific *Sometimes the satellite's second path is too close to the first path, and the pictures snapped are duplicates. Therefore, whenever it begins a new path, the satellite compares pictures being snapped on this path with old pictures. The satellite scans its store of old pictures and looks for a match with pictures being snapped.*

Validation of materials. To verify that the specific and abstract versions of each passage embodied the same relational structure, the two versions were given to two groups of subjects (n=14,14) not involved in the main experiment. Half of each group received one passage from each analogy. Subjects were also given abstract statements, some of which characterized the relational structure common to the base and target of each analogy. Other statements served as distractors.

These subjects' task was to identify which abstract statement belonged to which passage. Following this, subjects were given the intended match and they rated how well the abstract structure applied to the passage. As expected there was no difference between groups in the mean number of correct matches made (the means were 4.57 and 4.43 for the specific and abstract passages respectively). Also, there was no difference in the ratings (the mean rating was 6.2 in each group). These results suggest that the specific and abstract passages did not differ in how well they expressed the relational structure common to analogous pairs.

Procedure

Learning session. For each of five base passages, subjects: (a) studied the passage for three minutes, (b) tried to recall the passage in writing; subjects were asked to recall a passage "verbatim, exactly as it was written", and (c) corrected their recall by checking it against the passage. Subjects were asked to recall the passages verbatim to increase the likelihood that their representation of the passages would correspond to the particular type of passage read (abstract or specific).

Analogy session. Two days after the learning session subjects received six new passages, five of which were analogous to the five base passages.

1. Subjects first received an Access Task. Subjects were allowed three minutes to read each new passage and name an analogous passage from the Learning Session. (An analogy was defined for subjects as a partial similarity, and two examples of analogies with relational similarities were given.) Subjects were allowed to think of more than one analog for a given target.

2. Next, in a Recall Task subjects were asked to describe from memory each of the five base passages they learned during the Learning Session. The purpose of this task was to provide evidence that the specific, abstract and mixed passages were comparable in terms of how well they were retained.

3. Finally, in a Mapping Task subjects were told which passages formed an analogous pair. For each pair they were to describe how the passages were analogous.

Results

Access Task

Subjects were scored for the number of correct base- target matches made during the Access Task (possible scores were 0 to 5). Figure 1 shows the mean number of correct matches made in each group. The Abstract group made significantly more correct matches than the Specific group. ($F1,52 = 13.1$, $p < .001$). Performance of the Mixed group is intermediate and does not differ significantly from either that of the Abstract or Specific groups.

Figure 2 shows the proportion of subjects in each group making the correct match for the individual analogies. For four of the five analogies, significantly more subjects made correct matches in the Abstract group than in the Specific group (Fisher exact tests, $p <.001,.001,.01, .05$).

Analyses of errors revealed that subjects in the Specific group were more likely than subjects in the Abstract or Mixed groups to match passages on the basis of surface similarities. For example, they more frequently paired the Tams with the other passage concerning fierce creatures. The mean number of surface matches in the Specific, Abstract and Mixed groups respectively is 2.00, 1.29 and 1.36. The difference between the Specific and Abstract group is significant ($F_{1,52} = 4.02$, $p < .05$).

Recall Task

Subjects' recall of each base passage was scored for the presence of components of the analogous relational structure. Overall there was no significant difference between groups (the mean proportion of components present is .50, .43 , and .43 in the Abstract, Specific and Mixed groups respectively). However, the Abstract group did show significantly better recall than the Specific group for two of the passages, and significantly better recall than the Mixed group for one passage. These results raise the possibility that poor retention of the base passages among Specific subjects may have influenced the difference between groups in the Access Task. On the other hand, carry-over effects from success on the Access Task may have improved the recall scores of Abstract subjects. This issue is addressed in a follow-up experiment.

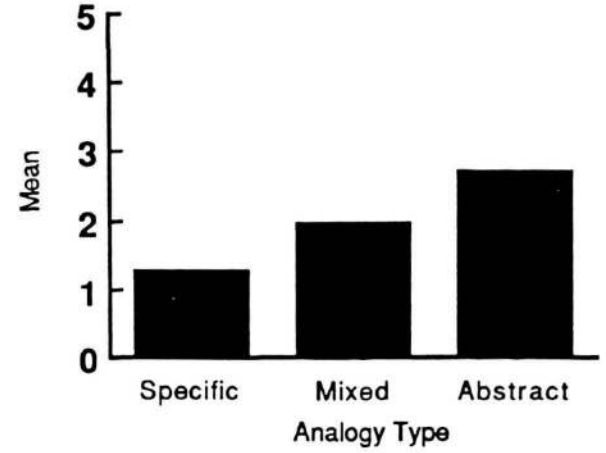


Figure 1 Access Task: Mean Number of Correct Matches in each Group

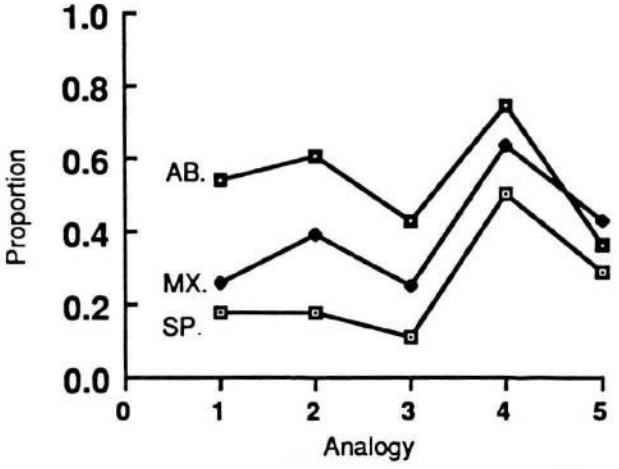


Figure 2 Access Task: Proportion of Subjects in each Group Making the Correct Match for each Analogy

Mapping Task

Subjects' descriptions of the similarity between base and target passages were scored for the presence of components of the analogous relational structure. The mean proportion of components mapped across the five analogies did not significantly differ between groups (M= .5, .42 and .43 for the Abstract, Specific and Mixed groups respectively). When the individual analogies are considered, the Abstract group showed significantly better mapping than the Specific group for one analogy only. No other differences are significant. Thus, although the abstract materials allowed better access of a base domain in the presence of the target, they did not have a strong effect on

mapping the structure of a base and target presented for comparison.

Follow-up Experiment

A follow-up experiment was conducted to examine the possibility that the abstract passages were easier to retain in memory than the specific or mixed passages. Three groups of subjects (n=17,16,17) received either abstract, specific, or mixed passages in a Learning Session identical to the Learning Session used in the main experiment. Subjects returned for a second session after two days. During the second session subjects recalled in writing the passages they had learned.

Responses were scored for the presence of components of the relational structure embodied in the passages. There was no evidence that the different types of passages were more or less easy to retain. The mean proportion of components recalled was .47, .51 and .46 in the Abstract, Specific and Mixed groups respectively. Also, when individual passages are examined, no differences between groups are found. These results suggest that the group differences in access of analogs found in Experiment 1 cannot be attributed to differences in retention of the different types of passages.

General Discussion

Analogical transfer between disparate domains is often difficult because the structural similarities between domains are difficult to detect. The common relational structure is embedded in information which is dissimilar at the surface level. Thus, recognizing analogical similarity often requires transforming domain representations into a more compatible form. For example analogous relations may have to be generalized to a higher level of abstraction or decomposed until a common component is found. The present research attempted to provide domain representations for subjects that should make structural similarity more directly detectable. Specifically, within a domain the analogous relations were expressed in either a relatively domain-specific or domain-general form. The relations were always embedded in a particular context, but the relations themselves were expressed with more or less abstract terms, and more or less detail was given about how the relations were instantiated. These manipulations should affect whether relations in one domain are characterized in a manner compatible with a different domain. This aspect of domain representations appeared to affect analogical access. Abstract subjects were better than Specific subjects at accessing a base domain in the presence of the target. Specific subjects often failed to notice an analogical match, and unlike the Abstract subjects, they tended to make matches based on surface rather than structural similarities.

The locus of the difficulty for the Specific Group may have been (a) the absence of more general or higher-level terms to characterize relations, or (b) interference from the domain-specific detail. The intermediate performance of the Mixed group provides some evidence that both factors were important. For this group, the abstract descriptions may have helped, but the specific detail interfered with recognition of analogies. These subjects may have been poor relative to Abstract subjects because the detail may have dominated the representations and made the abstract concepts less salient.

The variation in domain descriptions affected access of analogs more than mapping information between already identified analogs. Both access and mapping require arriving at a compatible representation of analogs. However, because access requires search of LTM, factors that affect the efficiency at arriving at this representation may be more important for access. Subjects in the Specific group may have been relatively inefficient at detecting relational similarities. Mapping between domains already in working memory may demand less efficiency. Thus even with relatively specific domain representations, subjects could often eventually work out a compatible representation.

Different levels of generality of a knowledge structure may be appropriate for different uses. For literal categorization, relatively specific knowledge structures may be the most effective. For example, to improve recognition of instances of a concept, explanation based learning systems (e.g. Mitchell, Keller, & Kedar-Cabelli, 1986) translate a highly abstract concept representation into a

more specific one. This facilitates identification of cases that instantiate the concept in similar ways. Unlike literal categorization, analogy requires a disregard of the particular way structures are instantiated. However, this does not entail that the detection of analogs requires the induction of an explicit, domain-independent schema. Induction of a schema is often seen as the last stage of analogical processing (e.g. Gentner, 1989). One way to understand how analogical access may occur is to consider that even though analogous relations are linked to a particular context, their representation within that context may be relatively abstract. The present research suggests that representations that are relatively sparse and abstract may permit access of analogs from disparate domains.

References

Bassok, M., & Holyoak, K. J. (1989). Interdomain transfer between isomorphic topics in algebra and physics. Journal of Experimental Psychology: Learning, Memory, & Cognition, 15 , 153-166.

Chen, Z. & Daehler, M. W. (1989). Positive and negative transfer in analogical problem solving by 6-year old children. Cognitive Development, 4, 327-344.

Clement, Catherine (1987). Applying general principles to novel problems as a function of learning history: Abstraction from examples vs. studying general statements. In the Proceedings of the Ninth Annual Conference of the Cognitive Science Society. Hillsdale, NJ: Erlbaum.

Faries, J. M. & Reiser, B. J. (1988). Access and use of previous solutions in a problem solving situation. In the Proceedings of the Tenth Annual Conference of the Cognitive Science Society. Hillsdale, NJ: Erlbaum.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. Cognitive Science, 7, 155-170.

Gentner, D. (1989). The mechanisms of analogical learning. In S. Vosniadou & A. Ortony (Eds.) Similarity and analogical reasoning. London: Cambridge University Press.

Gentner, D. & Landers, R. (1985). Analogical reminding: A good match is hard to find. In Proceedings of the International Conference on Systems, Man and Cybernetics. Tucson, AZ.

Gick, M. L. & Holyoak, K. J. (1980). Analogical problem solving. Cognitive Psychology, 12, 306-355.

Gick, M. L. & Holyoak, K. J. (1983). Schema induction and analogical transfer. Cognitive Psychology, 15, 1-38.

Holyoak, K. J. & Koh, K. (1987). Surface and structural similarity in analogical transfer. Memory and Cognition, 15, 332-340.

Lewis, M. W. & Anderson, J. R. (1985). Discrimination of operator schemata in problem solving: Learning from examples. Cognitive Psychology, 17, 26-65.

Mitchell, T. M., Keller, R. M., & Kedar-Cabelli, S. T. (1986). Explanation-based generalization: A unifying view. Machine learning, 1, 47-80.

Novick, L. R. (1988) Analogical transfer, problem similarity, and expertise. Journal of Experimental Psychology: Learning, Memory, & Cognition, 14, 510-520.

Rattermann, M. J. & Gentner, D. (1987). Analogy and similarity: Determinants of accessibility and inferential soundness. In the Proceedings of the Ninth Annual Conference of the Cognitive Science Society. Hillsdale, NJ: Erlbaum.

Reed, S. K., Ernst, G. A., & Banerji, R. (1974). The role of analogy in transfer between similar problem states. Cognitive Psychology, 6, 436-450.

Ross, B. H. (1984). Remindings and their effect in learning a cognitive skill. Cognitive Psychology, 16, 371-416.

Ross, B. H. (1987). This is like that: The use of earlier problems and the separation of similarity effects. Journal of Experimental Psychology: Learning, Memory, & Cognition, 13, 629-637.

Seifert, C. M., McKoon, G., Abelson, R. P., & Ratcliff, R. (1986). Memory connections between thematically similar episodes. Journal of Experimental Psychology: Learning, Memory, & Cognition, 12, 220-231.

Acknowledgements

A paper based on these data was presented at the meetings of the Psychonomics Society, Atlanta, GA. November 1989. We thank Maureen Clifford, Minh Thu Ti Le, Dedre Gentner and Ronald Mawby.

What Should I Do Now?
Using Goal Sequitur Knowledge to Choose the Next Problem Solving Step [1]

Michael Redmond
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, Georgia 30332-0280
E-mail: redmond@pravda.gatech.edu

Abstract

Many problems require multi-step solutions. This is true of both planning and diagnosis. How can a problem solver best generate an ordered sequence of actions to resolve a problem? In many domains, complete pre-planning is not an option because the results of steps can vary, thus a large tree of possible sequences would have to be generated. We propose a method that integrates the use of previous plans or cases with use of knowledge of relationships between goals, and the use of reasoning using domain knowledge to incrementally suggest the actions to take. The suggestion process is constrained by heuristics that specify the circumstances under which an instance of a particular reasoning goal can follow from an instance of other reasoning goals. We discuss the general approach, then present the suggestion methods and the constraints.

1 Introduction

There are many problems for which a multi-step solution must be generated. Such problems occur both in planning and in diagnosis. For instance, in automobile troubleshooting, a possible sequence of actions includes clarifying the complaint, verifying the complaint, generating hypotheses in some order, testing hypotheses, interpreting the test results, carrying out repairs, and testing the repairs. Complete pre-planning of troubleshooting steps may be inefficient. The number of possible choices and the variety of possible results of the actions can lead to a large, very bushy tree of possible paths. Generation of the complete trouble-tree for the given car and problem would be an expensive task to do, and might not even be possible. In addition, one action may not directly follow from the previous action. The question raised is how to best generate an ordered sequence of actions to resolve a problem.

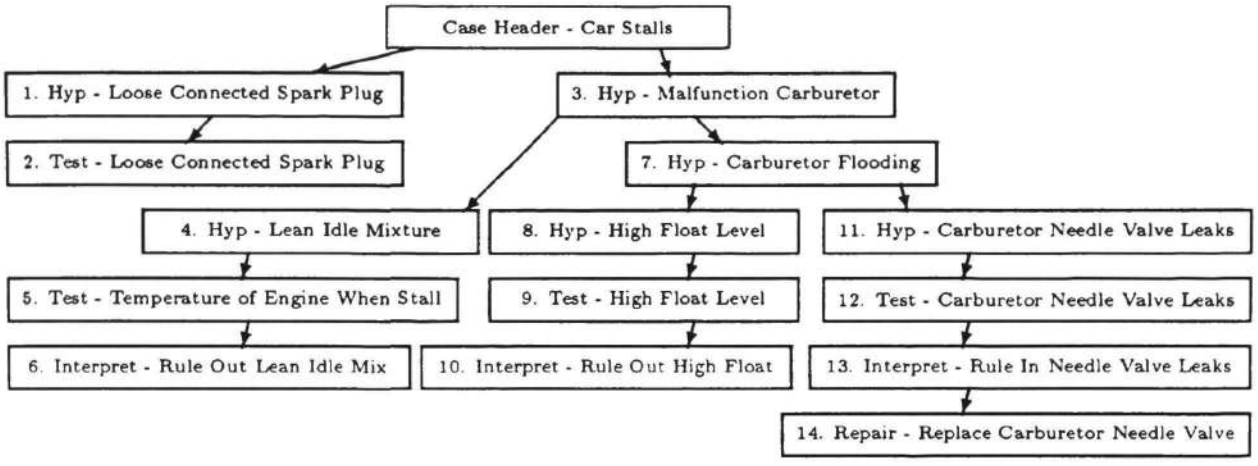
For example, for a stalling car, a possible sequence of actions is shown in Figure 1. The problem solver first hypothesizes a loose spark plug and a test of the hypothesis finds it not to be true. The problem solver hypothesizes that the carburetor is malfunctioning, then refines that guess to the more specific hypothesis of the idle mixture being lean. This is tested and the result suggests that the idle mixture is probably not the problem. Next, the problem solver generates a hypothesis that the carburetor is flooding, refines that to a hypothesis that the float level has become set too high, and tests for that. However, the test result indicates that that is not the problem. The problem solver generates another refinement, that the carburetor needle valve is leaking, allowing fuel in when it should not. This is tested, and is found to be true. A repair is done, and is tested, and results in the elimination of the problem.

These actions are not independent. Results of early actions influence future actions, and results cannot be predicted with certainty. If the spark plugs turned out to be loose, a repair would be done at that point and the problem solving would be complete if that is the only problem. If the needle valve is not leaking, further steps would be necessary beyond those in Figure 1. If, as a by-product of the test of the float level being high, it was determined that the fuel level in the carburetor was not too high, a different hypothesis would have been pursued opportunistically, instead of continuing to pursue the carburetor flooding hypothesis.

This example illustrates several points. First, it shows a situation where complete pre-planning would be inefficient. Not only can the number of possible choices and the variety of possible results of the actions lead to a large, very bushy tree of possible paths, but in many problems much of the tree would not be used. Second, it illustrates that when choosing the next action to take, the next action may not follow from the

[1] This research has been supported by the Army Research Institute for the Behavioral and Social Sciences under Contract No. MDA-903-86-C-173, and by DARPA contract F49620-88-C-0058 monitored by AFOSR. The author wishes to thank Janet Kolodner for her advice and guidance, and Tom Hinrichs, Steve Robinson, and Joel Martin for helpful comments on earlier versions of the paper.

most recent action. For example, step 7 follows from step 3. In addition, following a trouble-tree would not suggest taking advantage of unexpected opportunities, such as following up of the by-product of a test, as seen above.



The nodes represent the different goal instances that have been pursued. They are numbered in the temporal order in which they occurred. The links represent the relationships of which goal instance followed from which goal instance.

Figure 1: A Multi-step Solution in Automobile Troubleshooting.

The problem to be addressed in this paper is three-fold:

1. How can a problem solver efficiently generate successive goals and actions in a multi-step solution?
2. What knowledge is needed to generate the succeeding goals and actions?
3. How should the generation process be controlled and suggestions selected?

We address this problem in the task domain of automobile troubleshooting. Our program, CELIA (Cases and Explanations in Learning; an Integrated Approach), solves problems by generating and achieving reasoning goals. As in the example above, often later goals cannot be generated until earlier ones have been achieved. In addition, the program learns by understanding and explaining the statements and actions of a teacher.[2] The same process that generates goals during problem solving generates them during learning, where they act as expectations of what the teacher will do or say next.

As we will show, our approach integrates the use of four important types of knowledge to generate goals: previous solutions or cases (as in *Case-based Reasoning* [Kolodner and Simpson 1984]), knowledge of relationships between reasoning goals (in this case, knowledge of the troubleshooting process), and causal knowledge, including structural and functional knowledge of the domain. We will discuss generation of suggested actions, and control of the process, and then present an example.

2 Generation of New Subgoals and Actions

We have found four types of knowledge useful for generating subgoals and actions:

1. Case knowledge.
2. Causal knowledge of components, mainly functional knowledge.
3. Structural knowledge of component parts, including part/whole and adjacency relationships.
4. Knowledge of how to do the task, or the relationships between reasoning goals.

[2]Not natural language.

Case Knowledge: Case-based Reasoning (CBR) is a method of using previous episodes to suggest solutions to new problems. CBR is an important problem solving technique because it allows a reasoner to solve problems efficiently when previous similar experiences are available and complete knowledge is not present. In this type of problem, a case provides an ordered set of actions that have worked in the past. Thus remembering a part of a case, or *snippet*, suggests the next action. A next action can be suggested by the case the reasoner is currently reasoning from, or if context changes, by another case that becomes more relevant.

Causal knowledge: A next action or subgoal can also be suggested by causal knowledge. A causal link from a previous action can make the suggestion. Some aspect of the previous goal, for example the test done, the test result, the hypothesis generated, the fix done, something ruled out, is the starting point for the reasoning. The reasoning can proceed from that point forward toward effects of that aspect, or backward toward causes of that aspect.[3] The furthest point of progress in the causal reasoning is suggested as the value for the instance of the reasoning goal to be suggested. The causal reasoning uses both functional and structural knowledge of the domain.

Structural knowledge of components: Part/whole and topological knowledge of components involved in a previous action can suggest the next action. For example, a problem with the electrical system might be due to a problem with the battery. If the most recent instance of a generate hypothesis goal was that the electrical system is faulty, the next appropriate goal instance might be the generation of a hypothesis that the battery is faulty.

Knowledge of how to solve problems in the domain: How problems are normally solved is also important to generating goal and action sequences. Our method uses heuristic knowledge in the form of a set of the types of reasoning goals, or *goal types*, which can follow from each reasoning goal type, called *goal sequitur* knowledge, or sequitur knowledge for short.[4] [5] For example, hypothesis generation goals can be followed by tests of hypotheses, or by further hypothesis generation.

In general, there are many possible succeeding goals a problem solver might generate at any time. Good problem solvers generate goals that can lead them toward their final destination in the most opportune way. In the remainder of this paper we will present a way to choose the next goal or step wisely. We will show that the fourth type of knowledge listed above, knowledge of the problem solving task, is primary to this endeavor, providing guidance for moving toward a solution.

Our method uses three main types of heuristics for this task: *suggestor* heuristics suggest new steps, *restrictor* heuristics constrain the behavior of the suggestors, and *selector* heuristics choose the best of the suggested next steps.

We begin by presenting the four main types of suggestor heuristics:

1. Case Sequential Access
2. Case Direct Access
3. Causal link
4. Refinement (Part/Whole)

The suggestor heuristics, or suggestors, indicate ways to generate possible instances of the consequent goal type.[6] Running these heuristics results in a set of possible succeeding reasoning goals and actions.

[3] Reasoning is uncoupled from the reasoning goal involved. Given a hypothesis that the fuel mixture is too lean, reasoning proceeds from the state that the fuel mixture is too lean (which may or may not be true), not from the *hypothesis* that the fuel mixture is too lean. Thus the causal reasoning does not depend on the reasoning goals involved in the domain, or vary depending on those involved.

[4] *Goal types* are general types of reasoning goals such as clarifying the complaint, verifying the complaint, generating hypotheses, testing hypotheses, interpreting the test results, carrying out repairs, and testing the repairs. They could also be considered subtasks of troubleshooting. Goal instances are specific instantiations of goal types, such as the specific test used to test a specific hypothesis.

[5] As *non sequitur* means an inference or conclusion that does not follow from established premises or evidence, we use sequitur to refer to a goal or action that follows from a previous goal or action.

[6] The borrowing of the logical terms antecedent and consequent should not be taken as an indication that the second goal logically follows from the first. The consequent only plausibly follows from the antecedent.

2.1 Case Suggestors

As noted above, a case provides an ordered set of actions that have worked in the past. Thus remembering a case appropriate to the current context suggests the next action. Multiple parts of multiple cases can be useful in solving a particular problem. Useful parts can be accessed directly, by retrieving the relevant part of a relevant case, or sequentially, by continuing to follow a previous case while it continues to be relevant. The two suggestors that use parts of previous cases are based on these two methods.

2.1.1 Sequential Access

If the results of running a step in the new situation match those obtained when it was run in the previous case, the next step in sequence in that case can be suggested. The **Continue-following-link** suggestor does this.

2.1.2 Direct Access

If the results are different, however, the **Case-snippet** suggestor uses direct access to part of a different case that can provide a suggestion of what to do next. Retrieval involves matching the current situation to the case part, or snippet's goal and context. In our system, CELIA, retrieval via direct access is first restricted to snippets that are centered around the goal type being considered. Then a weighted similarity metric is used, with matching occurring for all features within the context. The context includes the internal context, the results of actions taken up to that point in problem solving, so the retrieved piece is influenced by the results of goals pursued so far in this problem.

2.2 Causal Link Suggestors

Causal link suggestors use domain knowledge of function and structure to reason either forward or backward from a clause in the preceding goal instance in order to suggest the main clause for the consequent goal instance.

Variations in these heuristics include:

- Whether reasoning is forward or backward from the initial clause. For instance, reasoning backwards can lead toward suggesting hypotheses that could be root causes. Reasoning forward can lead to suggesting tests of hypotheses based on their potential effects.
- Which aspect of the previous goal instance to use as the initial clause. For instance, when reasoning from a test of a hypothesis a useful starting point is the test result. When reasoning from the interpretation of a test useful starting points include things ruled in or ruled out.
- Whether the initial clause is returned as a result when no progress is made in the causal chaining. When the consequent goal type is the same as the antecedent goal type, this is not appropriate.
- Whether to return a contradiction of the linked clause, or just the linked clause. For instance, a test for a contradiction of something that follows from a hypothesis can be a good test of the hypothesis.

2.3 Refinement Suggestors

Refinement suggestors use part/whole knowledge to suggest a new goal instance through refinement of the preceding goal instance. Either

- Its component is below the previous goal instance's component in the partonomy, (a *leak in the fuel line* is more refined than a *leak in the fuel system*). The previous goal instance's component is refined to a component that is part of the previous component. or
- The component is the same and the new predicate is more specific, (the *ECM not being grounded properly* is more refined than a *malfunction in the ECM*). This requires use of knowledge of the functions of the involved components. If the predicate is 'malfunction', then those predicates that are involved in obstacles to the component's function are considered as refinements of the previous predicate.

Variations in these heuristics include:

- Whether a clause that is equally as refined is acceptable. When the consequent goal type is the same as the antecedent goal type, this is not appropriate.
- Which aspect of the previous goal instance to use as the initial clause. For instance, refining the interpretation of a test, useful starting points include things ruled in or ruled out.

3 Controlling Suggestions: Restrictors

There are, in general, large numbers of possible next steps that could be generated by the methods above. Restrictors constrain the suggestion process so that effort is not expended trying to generate actions in directions that will not prove fruitful. In general, restrictors rule out goal sequences that are sometimes possible, but are not appropriate in the particular current circumstance. For instance, a test should not follow from a hypothesis that has already been tested, and a hypothesis should not follow from a hypothesis that has already been tested. However, a hypothesis can follow from a hypothesis that has already been refined, it could be another refinement. These examples suggest two of the restrictor heuristics.

No-sibs restricts a goal following from a previous goal to contexts in which no action has already followed from the antecedent.

Only-same-type-sibs restricts a goal following from a previous goal to contexts in which either no action has already followed from the antecedent, or contexts in which only actions fulfilling the same goal type have already followed from the antecedent.

Because some goal types should only follow from the most recent instance of some other goal types, restrictors are necessary for that purpose. For example, an interpretation of a test should follow from the most recent test instead of some previous test. This is clearly not the case for all goal sequences. For example, a number of hypotheses could be advanced, then tested in order, thus the test would not follow from the most recent hypothesis.

Most-recent restricts a goal following from another goal to contexts in which the previous goal was the most recent instance of that goal type.

Also needed are restrictors that constrain what can follow from the interpretation of a test. After a test result has been interpreted, what follows depends on the interpretation. If the hypothesis that is being pursued has been ruled in, either the hypothesis can be refined further, or a repair can be made. If nothing has been ruled in, and something ruled out, it is possible that the complaint should be further clarified. The following two restrictors are used.

Prev-ruled-out restricts a goal following from a previous action to contexts in which the previous actions included ruling out some condition.

Prev-ruled-in restricts a goal following from a previous goal to contexts in which the previous actions included ruling in some condition.

4 Selectors

Even with restriction, several steps might plausibly follow the current situation. Selector heuristics choose the best of those generated. Selectors work in two stages. Before suggestors are run, some selectors specify allocations of computational effort to the different suggestors associated with each of the possible future goal sequences. Then, after generation of plausible next steps, the best next step is chosen based on the rest of the selectors, and the amount of the allocation used.

The selectors that influence allocations include:

1. Allocate more effort to generating possibilities following from more recent goals pursued.
2. Allocate more effort to generating possibilities following from a leaf node of problem solving.
3. Allocate more effort to generating possibilities following from a problem solving node closer to the most recent goal pursued.

These allocations serve to limit the processing suggestors can do before cutting off search. This is important because it keeps the slowest heuristics, such as causal chaining, which can be intractable, from slowing the process down too much.[7]

When suggestions have been made, the choice of what goal and action to take is based on the percentage of allocated effort used in conjunction with the following preferences:

1. Favor possibilities generated using continue-following-link, then case-snippet, then other methods such as causal chaining.
2. Favor possibilities generated using goal sequences judged more likely in our analysis of the diagnostic task.
3. Favor possibilities generated from reasoning goals with more restrictor heuristics. These are less likely to have inadvertently escaped restriction.
4. Favor possibilities generated from reasoning goals with fewer suggestor heuristics. These are less likely to be low quality 'shots in the dark'.

The combined effect of the selectors is to favor continuing following along from the most recent goal, using a previously retrieved case snippet, or a newly retrieved case snippet. The preference is not absolute, however. It does not make the easiest suggestor heuristics dominant, because the allocated effort can vary widely based on the factors discussed above.

5 Example

We will illustrate the process of choosing the next action using the example shown in Figure 2. This is an English-ized version of a sequence of problem solving steps generated by our program CELIA.[8] The problem solver first clarifies the complaint, then verifies the complaint to make sure that the problem can be recreated. The problem solver hypothesizes that the carburetor is malfunctioning, then refines that guess. The idle speed is considered, and rejected. The idle mixture is considered, tested, and repaired, and yet the problems remain. A further hypothesis of the throttle dashpot being out of place is generated, and tested.

1. Clarify the complaint
2. Verify the complaint
3. Generate a hypothesis - carburetor malfunction
4. Generate a hypothesis - low idle speed
5. Test hypothesis - temperature of engine when stalling occurs (warm)
6. Interpret Test - idle speed not a problem; idle mixture possible problem
7. Repair - Adjust idle mixture screw
8. Test Repair - engine still stall? (yes)
9. Interpret Test - idle mixture not the problem
10. Generate a hypothesis - throttle dashpot out of place
11. Test hypothesis - distance between throttle dashpot stem and throttle lever small? (no)

Figure 2: Example Multi-step Solution in Automobile Troubleshooting.

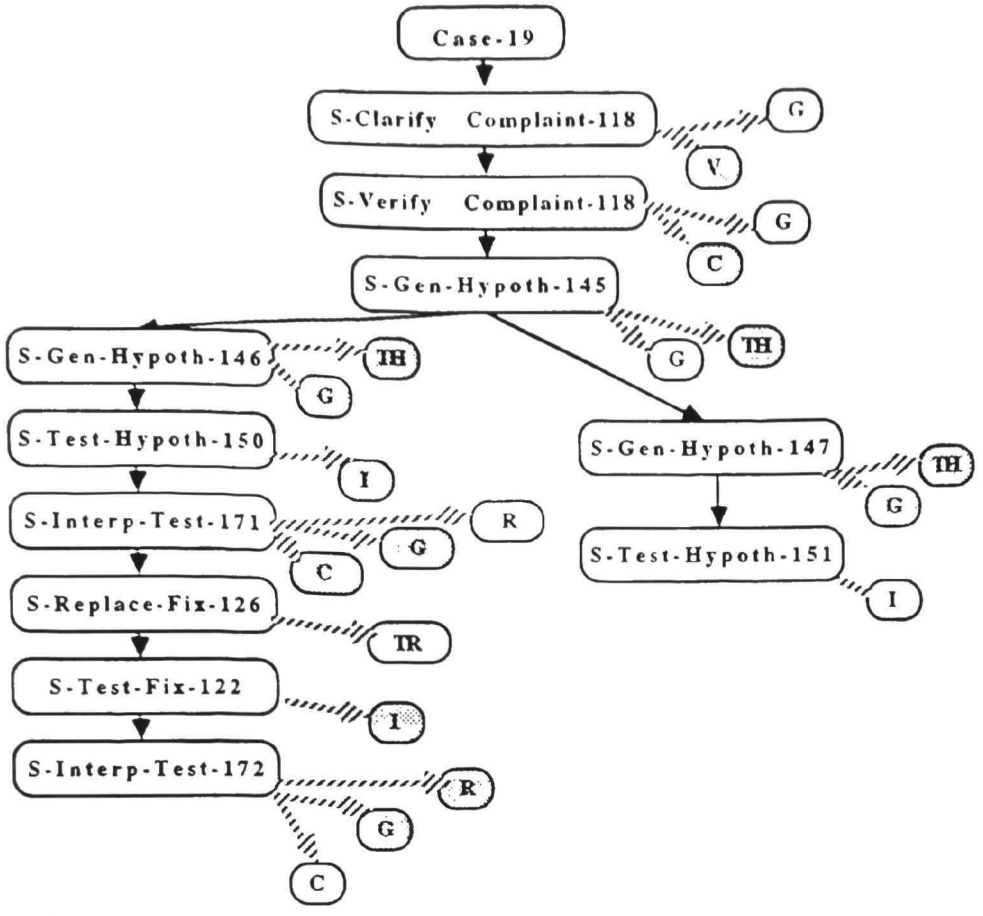
After step 11, the problem solving can be illustrated by the large nodes of the tree shown in Figure 3. Nodes represent goals that have been pursued so far. Links represent the sequencing relationships between the goals. S-Verify-Complaint-118 corresponds to the first step, S-Gen-Hypoth-145 corresponds to step 3 - Generating a hypothesis, in this instance a carburetor malfunction. The most recently completed action is included in S-Test-Hypoth-151. At this point the next action must be generated.

The small ovals in Figure 3 show types of possible succeeding goals that can follow from the parts of the problem solving to this point. The key for the different goal types is given.

The set of possibilities can be reduced significantly using the restrictor heuristics. Shaded ovals in Figure 3 show the effects of the restrictors. These are the directions restrictors have determined not to be fruitful.

[7] Causal chaining is constrained both by the strategy of trying to form a connection between actions rather than trying to form a connection over the large space between complaints and root causes, and by selection allocations.

[8] Actually, it was generated as predictions of what an expert would do by the learning component of CELIA using the same methods as described for the problem solving component. The problem solving component has not yet had the equivalent upgrade from the previous version.



The nodes represent the different goals that have been pursued so far. The links represent the sequitur relationships between the goals. The small ovals connected to nodes with striped arrows represent the possible goal types that can follow from the goal types of the nodes.

Figure 3: Remaining Possible Next Goal Types after restriction.

For each of the remaining possibilities there are several applicable suggestors, these are able to generate 26 possibilities including:

Goal Type	Instance	Following from Piece	Using Suggestor
G-Interp-Test	(Incorrect (Position Throttle-Dashpot))	S-Test-Hypothesis-151	Case-snippet
G-Gen-Hypoth	(High (Contains Carburetor-Float-Bowl Fuel))	S-Verify-Complaint-118	Case-snippet
G-Test-Repair	(Small (Dist Throttle-Dashpot-Stem Throttle-Lever))	S-Replace-Fix-126	Case-snippet
G-Test-Repair	(Low (Position Idle-Mixture-Screw))	S-Replace-Fix-126	Un-Improve
G-Test-Repair	(Increase (Position Idle-Mixture-Screw))	S-Replace-Fix-126	Equivalent
G-Replace-Fix	(Lean (Position Idle-Mixture-Screw))	S-Interpret-Test-171	Fault-Determination
G-Gen-Hypoth	(Hole Carburetor-Barrel)	S-Gen-Hypoth-145	More-Refined
G-Gen-Hypoth	(Clogged Carburetor-Pipe-To-Venturi)	S-Gen-Hypoth-145	More-Refined

From among these, the selector heuristics choose the first action, the interpretation of the test ruling out the hypothesis of the throttle dashpot being out of place. This suggestion was chosen due to several factors:

1. It was generated from the most recent previous goal and actions. Therefore, the suggestor which generated it was allocated a high amount of processing, of which not much was used in retrieving the case snippet that suggested the interpretation.
2. It was generated from a case snippet. Therefore it was favored at selection time.
3. Tests (of hypotheses or of repairs) need to be interpreted, so the judged likelihood of an interpretation of a test following from a test of a hypothesis is high, favoring this suggestion.

6 Related Work and Conclusions

A number of other efforts share some flavor with our approach. Koton [1988] combines use of a number of reasoning methods. First, associations formed from generalizations of cases are tried, then cases, and lastly model knowledge. However, the strict ordering of methods used is less flexible. More importantly, her approach does not generate steps for a multi-step solution, but rather a classification. Carbonell [1986] generates steps for a multi-step solution using a previous case. Domain knowledge is used in adapting the solution, but *one* case will either provide a whole solution or have to be abandoned or adapted. Parts of multiple cases cannot be used. Allen and Langley [1989] generate multi-step solutions using a combination of generalizations, cases, and domain knowledge (in the form of operators). However, they do not retain relations between problems and subproblems, so their DAEDALUS system cannot use an entire previous plan from memory.

Our approach combines the use of several types of knowledge and reasoning techniques. It takes advantage of knowledge about the relationships between goal types to provide constraint on the problem of coming up with the next action to do. The problem solving is flexible and can take advantage of the results of previous actions when deciding what to do next, while remaining goal directed. The approach has three phases – restrictors limit the number of possibilities to be considered, suggestors generate possible next actions, and selectors chose the action to take. There are several advantages to the approach. It combines multiple reasoning methods in a flexible manner. Problem solving is flexible enough to use whatever knowledge is available, using cases when appropriate cases can be found, domain knowledge when it can be useful. It is a flexible way of using parts of multiple cases in forming a solution that is a synthesis of steps. Problem solving can change directions when the results of the problem solving make that necessary. A major side benefit is that many of the suggestor heuristics can benefit when further knowledge is added to the system, in the form of new cases or new domain knowledge. Our system, CELIA, is a learning system, and is designed to acquire such knowledge. This makes problem solving more effective without having to learn new heuristics.

References

Allen, J. A. & Langley, P. (1989). Using concept hierarchies to organize plan knowledge. In *Proceedings of the Sixth Annual International Workshop on Machine Learning.*

Carbonell, J. (1986). Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In Michalski, R., Carbonell, J., & Mitchell, T., (Eds.). *Machine Learning: An Artificial Intelligence Approach, Volume II.* Morgan Kaufmann, Los Altos, CA.

Kolodner, J. & Simpson, R. J. (1984). A case for case-based reasoning. In *Proceedings of the Sixth Annual Conference of the Cognitive Science Society*, Hillsdale, NJ. Lawrence Erlbaum Associates.

Koton, P. (1988). *Using experience in learning and problem solving.* PhD thesis, Massachussetts Institute of Technology, Cambridge, MA.

Lancaster, J. & Kolodner, J. (1987). Problem solving in a natural task as a function of experience. In *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, Hillsdale, NJ. Lawrence Erlbaum Associates.

Lancaster, J. & Kolodner, J. (1988). Varieties of learning from problem solving experience. In *Proceedings of the Tenth Annual Conference of the Cognitive Science Society*, Hillsdale, NJ. Lawrence Erlbaum Associates.

Redmond, M. (1989a). Combining case-based reasoning, explanation-based learning and learning from instruction. In *Proceedings of the Sixth Annual International Workshop on Machine Learning*, San Mateo, CA. Morgan Kaufmann.

Redmond, M. (1989b). Combining explanation types for learning by understanding instructional examples. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, Hillsdale, NJ. Lawrence Erlbaum Associates.

Redmond, M. (1989c). Learning from others' experience: creating cases from examples. In *Proceedings of the Second Workshop on Case-Based Reasoning*, San Mateo, CA. Morgan Kaufmann.

A Case-Based Approach to Creativity in Problem Solving[1]

Janet L. Kolodner & Theresa Louise Penberthy
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332-0280
email: jlk@gatech.edu, louise%pravda@gatech.edu

Abstract

One of the major activities creative problem solvers engage in is exploration and evaluation of alternatives, often adapting and merging several possibilities to create a solution to the new problem. We propose a process that models this activity and discuss the requirements it puts on representations and reasoning processes and present a program that solves problems by following this procedure.

Introduction

Often, when people solve problems, they create new solutions by adapting and combining previous solutions they have learned about or created themselves. Sometimes a new solution can be created by merely "tweaking" an old one in routine ways. Often, however, problem solving is less routine, requiring exploration of several alternatives, perhaps adapting and merging several possibilities gleaned from experience.

Consider, for example, a dietician in a nursing home planning meals for its residents. The normal routine is to come up with a standard breakfast and adapt it for those with special diets. For example, a patient with no teeth might be served runny eggs and ground-up toast. A more creative solution, however, might be to give the person without teeth a yogurt milkshake from time to time. The excerpt below shows how one person produced that solution.

```
adapting breakfast for a patient with no teeth,
standard way is to grind the toast in the blender, and leave
the eggs runny

==> evaluate: but that's boring, has lousy texture, and tastes bad.
what can someone without teeth eat that tastes better?

==> elaborate:  perhaps something liquid

breakfast, blender, liquid ==> instant breakfast

==> evaluate:  easy to make, texture right, but taste is
```

[1]This research is supported in part by DARPA under contract number F49620-88-C-0058, and monitored by AFOSR. Mike Redmond and Steve Robinson read preliminary drafts of the paper.

```
    uninteresting.  also, nutrition is wrong -- it should come from
    real foods rather than being artificially created

blender, liquid, real food, nutritious ==> yogurt milkshakes,
made with fruit juice, wheat germ, and yogurt (I drank them a lot as
supplements when I was pregnant)

==> evaluate:  easy to make, texture is good:  in addition to being
    liquid, there's a slight crunch;  nutrition might still be
    insufficient, might need more iron or protein for older people

..., extra nutrition ==> a friend who is a health nut
..., health nut ==> add brewer's yeast for extra protein and vitamins

..., extra nutrition, eating while pregnant ==>
blender, liquid, extra protein, pregnant, ... ==> add a raw egg

==> evaluate: can't use raw eggs any more because of possibility of
salmonella poisoning

==> solution:  yogurt milkshakes, with fruit juice, wheat germ,
yogurt, brewer's yeast

evaluate ease-of-prep:  easy
evaluate texture:  liquid, slight crunch
evaluate taste:  flavorful
evaluate nutrition:  good
```

This reasoning sequence is much like several other examples we have collected of creative problem solving. The reasoner starts with a partial, incomplete problem specification, and through a series of example retrievals and evaluations, eventually both defines the problem more clearly and creates a solution. Several solutions are considered, and the final one has elements of many considered before it. We refer to the process as **exploratory reasoning**.

Exploratory reasoning seems to have many of the elements of *brainstorming*. When brainstorming, a person or group of persons attempts to solve a problem by thinking of the most unusual, farfetched, and "off-the-wall" solutions they can. This helps them expand their thinking beyond the usual and routine, and to consider the problem in a new light. This process continues, as Shouksmith (Shouksmith, 1970) points out, until eventually the problem-solving efforts must persistently focus on refining a proposed solution until it is usable.

Our research goal, and the one reported on in this paper, has been to derive a model of the exploratory processes involved in creative problem solving. We are considering creative problem solving as an extension of more mundane reasoning. Creative reasoning, we think, should not require a different architecture or set of processes. Rather, common processes should be extended and packaged for creative reasoning.

The exploratory processes that we have observed in creative problem solving are similar to the processes involved in *case-based reasoning* (Hammond, 1989; Kolodner & Simpson, 1984; Simpson, 1985). In case-based reasoning (CBR), problems are solved by remembering old situations similar to a new one and adapting the solutions to those problems to fit the new situation rather than solving problems from scratch each time. More mundane case-based reasoning attempts to apply old solutions directly rather than trying out many alternatives, as seen in our protocols.

A case-based reasoner includes in its architecture a memory of cases, retrieval algorithms that can retrieve the best partially matching cases from memory on demand, an adapter that can adapt an old solution to fit a new situation, and a goal scheduler to keep track of the reasoning. Our proposal extends CBR in the following ways:

1. Many cases are considered; many solutions are proposed, and solutions are often made up of combinations of features from several cases.
2. Evaluation of proposed solutions is a primary process.
3. Problem solving is incremental; problem solutions as well as descriptions are updated based on evaluations.
4. Evaluation includes willingness to accept what might not be "right;" odd proposals are considered for what they can contribute rather than being disregarded outright because they won't work.

The process we propose for the exploratory part of creative reasoning is as follows:

- **Retrieve** a set of cases (initially, use the original problem specification as a guide)
- For each case:
  - **Evaluate** the solution proposed by the case for its applicability to the new problem
  - **Evaluate** the solution proposed by the case for its adaptability to the new problem
  - Based on evaluations, **update the problem solution** and
  - **update the problem specification** appropriately
- Repeat until a satisfactory solution is created or found

In the remainder of this paper, we discuss each of these processes, describing the ways it must be extended for exploratory problem solving.

Memory and Retrieval Processes

As we look at the examples of exploratory reasoning that we have collected, we notice that there is a broad selection of cases remembered. Some are routine and some are novel. Each addresses in some way the goals of the problem solver, but none was created with exactly the new goal in mind

(e.g., yogurt milkshakes have always been drunk with sound natural teeth). In other words, they address the problem but this new problem was probably not anticipated at the time those cases were experienced.

Memory retrieval theories espoused in case-based reasoning suggest that indices associated with cases stored in memory indicate what is important about a case, and in most theories, that they also direct and restrict search through the memory (Hammond, 1989; Kolodner, 1983; Schank, 1982). Attempts to designate what makes one case better than another for case-based reasoning suggest that the goal of the reasoner is most important in choosing a best case and, furthermore, that the better the goal of the new case matches one being pursued in the new situation, the better that case will be for case-based reasoning (Kolodner, 1989).

The remindings we see in examining our creative problem solving protocols suggest a few other things. First, the fact that the problem solver's current goal could not have been anticipated at the time an old case was recalled indicates that *a priori* or anticipatory indexing is not sufficient to explain retrieval. Second, the protocols suggest that there often will be no cases that address the primary goals of the reasoner. Third, it seems that stranger cases are preferred over more normal ones.

From the above analysis, several things can be inferred about memory and retrieval. First, retrieval needs to be more flexible than an "index-following" scheme. It needs to allow seeing an old case in a new light. Features that were not salient at the time a case was experienced might be important for retrieval. This means cases should be retrievable both on salient features (that probably were indexed) and on features that were not originally salient, and hence were probably not indexed. PARADYME (Kolodner, 1989) suggests one way to do this: all partial matches are retrieved by a parallel algorithm, and then a set of preference heuristics chooses the best of those, preferring those that match on features marked as salient but allowing other cases to be recalled if no cases with marked salient features are found, and preferring those that can be used to address the problem solver's current goal over others.

The second change these protocols suggest is that ranking of retrieved cases based on which can best address the reasoner's current goal may not always be appropriate. Brainstorming requires being open to remembering things that may not address the current goal but instead might address some higher level or sibling goal. More investigation is needed of the kinds of preferences necessary to rank cases for exploratory reasoning.

Third, there needs to be a way of ascertaining whether recalled cases and derived solutions are ordinary or special.

Evaluation of Applicability and Adaptability

The solutions to cases that are remembered during exploratory reasoning are evaluated for their applicability and adaptability to the problem at hand. When evaluating, the reasoner must ask several questions: whether the old solution or some part of it is applicable to the new situation; whether it or some part of it is obviously inapplicable; whether it or some part of it could be adapted to fit the new situation. When yogurt milkshake is considered as a solution in the protocol above,

for example, it is inserted into the solution in progress because it achieves many of the problems goals. Later reasoning and exploration assumes it to be part of the solution and aims at refining it to solve the problem better.

Evaluation of an old solution might also cause the reasoner to add additional constraints to the problem description. Some will be desirable, and therefore added to the problem description. Others will be seen to be poor, and ruled out in the problem description. When one of us was trying to decide what to do with leftover white rice, for example, she considered making fried rice. However, she rejected that possibility because she had had Chinese food recently, and preferred not to have it again. The problem specification was fixed to rule out any dish of Chinese cuisine as a result of that consideration. Conversely, when she considered making rice pudding, though she rejected that also because it was sweet, she decided to make a baked dish with milk. The problem specification was augmented to prefer a baked dish with milk, such as a casserole.

Evaluation, then, is a key to exploration, since its results are used to both further define a problem and create a solution. But any solution that is remembered could be quite large, and any part of it might be evaluated. The big issue is how to provide focus the evaluation procedures so that they ask appropriate questions of a proposed solution.

Based on an examination of the task of recipe creation, we can propose several places from which these questions derive. First is **function-directed** evaluation. In the recipe creation domain, the purpose is to create something that can be eaten. Thus some evaluative questions arise from the concept of edibility. It is important, for example, to examine the taste and appeal of a dish to see if it is edible. Another set of evaluative criteria are **constraint-related**. Does the proposed solution fit the problem specification? If a dish is to be vegetarian, for example, evaluation procedures must ask if it is vegetarian or could be made that way easily.

A third set are derived from reasoning that occurred in creating previous solutions. We will call them **derivation-driven**. Old solutions themselves provide a rich and important source of questions if the considerations taken into account in creating them are saved. Consider, for example, the task of trying to decide if tofu can be substituted for cheese in tomato tart (a cheese pie flavored with dijon mustard and garlic and with tomatoes on top). The reasoner must be able to evaluate the original ingredient, and determine which of its characteristics are shared by the proposed substitute, which are not, and whether the differences matter. One way the right evaluative questions can be derived is by recalling another case where tofu was to be substituted for cheese. Concerns in that case are likely to be concerns in the new one too. For example, if in the previous case the texture of tofu was compared with the texture of the original ingredient, the reasoner might then ask about texture in the current case.

A fourth set of evaluative questions are **outcome-related**. Some outcome features are well-known and might be asked all the time. Others might be derivable only from other cases. A failure in a similar dish, for example, causes the reasoner to ask whether a similar failure might occur in the new one. If such a failure might occur, repairs suggested by the evaluation of the failure in the previous case will be helpful in fixing the new problem specification or solution.

Of course, if evaluative questions derive from previous cases, then cases must maintain their adaptation or derivation history. They must know where their solutions came from, what was taken into account in creating them, and evaluations of their outcomes.

Updating the Solution in Progress and Problem Specification

Evaluation of old solutions provides guidelines for filling in the solution in progress. A portion of an old solution that fits the new constraints added to the progressing solution. More interesting, however, is that evaluation can also result in update of the problem specification.

In ill-defined problem solving situations, the problem solver may not have all the necessary information at the beginning of problem solving. In these cases, the problem specification must be treated as a dynamically-changing entity. The results of evaluation can help with this. Ruling out Chinese cuisine as a result of remembering and evaluating a Chinese dish provides one example of this. In another instance, remembering a dish that was hard to prepare caused the reasoner to add "easy-to-prepare" to the problem specification. Both of these are cases of refining or adding to the descriptors in a problem specification.

The most interesting examples of changing the problem specification, however, are those in which the goals of the problem solving seem to be changed as a result of a "good" solution that is found or created. For example, in the white rice problem (mentioned above), rice frittata, a breakfast dish, came to mind. While it would certainly use white rice, it is not a dinner dish. Rather than throwing it out because it violated the goals of the problem solving, she changed her mind about the goals: a breakfast dish would be acceptable. In this way, the opportunity to fulfill more important goals is not quashed by the inability to fulfill less important ones.

There are several questions that arise here. First, if essential features of the problem can be changed during problem solving, then how can the goodness of a solution be evaluated? Certainly not based entirely on the problem specification. Second, is it "legal" to change all parts of a problem specification while solving the problem, or are some parts sacred?

We have addressed these two issues by making an assumption that some parts of a problem specification are more sacred than others. To date, we differentiate **primary** from **secondary** specifications, where primary ones are less likely to be changeable than secondary ones. This designation plays three roles during problem solving: First, when retrieving examples from memory, those that match on primary features are considered better matches than those matching on secondary features. Second, when evaluating applicability of a solution, it is judged more applicable if it achieves primary goals than if it achieves secondary ones. Third, when the problem specification is being updated, secondary features are allowed to be updated if an otherwise good solution is found, but primary features cannot be changed without good reason.

Our Implementation

We've implemented a program that solves problems by exploring a library of alternatives, evaluating relevant ones, and based on those evaluations, respecifies its problem and creates a solution. It is based on a protocol of a person attempting to decide what to do with leftover white rice.

The program's initial problem is to come up with a dinner dish that uses leftover white rice. It begins by remembering three dishes: fried rice, yeasted bread with cooked rice baked into it, and rice pudding. It evaluates each suggestion. Fried rice is rejected because the program knew that

Chinese food had been served too recently. It adds a note to the problem specification that Chinese cuisine is ruled out. It is also reminded of rice fritata, a breakfast dish. Considering rice fritata, it notes that it is a breakfast dish rather than a dinner dish, but it does achieve the goal of using some of the rice, and it is well-liked, so the program decides rice fritata should be made for breakfast.

Because the fritata doesn't use all the rice, however, the program continues its reasoning. It considers and rejects the yeasted bread because it takes too much time. It adds a note to the problem specification that the solution should be quick to prepare. The rice pudding is rejected because it is sweet. but based on desirable characteristics of rice pudding (baked, casserole-like, uses milk), it finds macaroni and cheese. Macaroni and cheese doesn't use rice, but it can be adapted to rice and cheese, which does use rice. Together, these two dishes satisfy the program's original primary goal. It proposes making rice fritata for breakfast and rice and cheese casserole for dinner.

While the program is still in its formative stages, it does many of the things discussed earlier in the paper. As it retrieves candidate cases, it augments the problem specification based on its evaluation of the cases. At a minimum, it evaluates alternatives for the appropriateness of their main ingredients and preparation method. When cases it is evaluating "know" how they were created or why they were selected, it also asks evaluative questions based on previous concerns. It updates its solution in progress based on these considerations. Additionally, the problem specification is augmented with desirable features of each recalled case and constrained by undesirable features, as described above. This augmentation is not permanent; when another case is chosen as the basis for remindings, the problem specification is altered to fit it instead.

Discussion and Conclusion

Our claim has been that a primary component of creative problem solving is the process of exploring and evaluating alternative solutions, often merging several solutions into one as a result. Goel & Pirolli (1989) have also observed creative designers redefine and elaborate their problem specifications, explore alternatives, and merge possibilities. However, they do not propose processes. We have tried to make more concrete the processes for doing the things they have studied, and to show how already-known processes can be extended for these tasks.

A key aspect of the creative process is the ability to ask questions. Others studying creativity have made a similar claim. According to Schank & Leake (1986), creativity lies in the ability to ask creative questions, and to use those questions to apply a preset *explanation pattern* (XP) in an unusual way to come up with interesting answers. While we agree that the ability to ask creative questions contributes to creative solutions, we propose that they also allow the reasoner to modify the problem specification, an important task when solving open-ended problems.

The most obvious application of creative reasoning is in those domains that are seen as inherently creative: art, for example, or meal planning, or architecture, and so on. But the approach we have introduced here can help reasoners in many domains.

Any problem solver can reach an impasse as it works to solve a problem. Exploratory reasoning provides techniques to breach such an impasse. By elaborating the original problem specification in more than one way, a problem solver can find different paths to solutions.

In a domain where the problem may be poorly defined, perhaps incomplete or overconstrained, the exploratory process can help elaborate the specification of the problem and provide a more complete basis for problem solving.

There many issues that still need to be addressed, the most important of which, we think, is control of evaluative processes. While we have insight into the derivation of evaluative questions, we still don't know how a reasoner chooses among the many paths that reasoning could take.

An understanding of creative problem solving processes has several important implications. If we understand creative processes, which parts are hard and which are easy, we will be able to create the right kinds of tools to help problem solvers with their tasks. We will find out which processes can be relegated to a machine and which will need to continue to be done by people. This understanding will also help us in building the kinds of tools and developing the kinds of curricula that can best be used to train creative problem solvers of the future.

Bibliography

Goel, V. & Pirolli, P. (1989). Motivating the Notion of Generic Design Within Information Processing Theory: The Design Problem Space. *AI Magazine*, Vol 10, No. 1.

Hammond, K. J., *Case-Based Planning: Viewing Planning as a Memory Task.* San Diego: Academic Press; 1989.

Kolodner, J. L., Selecting the best case for a case-based reasoner. In: Proceedings of the Eleventh Annual Conference of the Cognitive Science Society, University of Michigan, Ann Arbor, Michigan, 1989.

Kolodner, J. L., and R. Simpson Jr., A case for case-based reasoning. In: Proceedings of the Sixth Annual Conference of the Cognitive Science Society. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1984.

Kolodner, J. L., Reconstructive memory: A computer model. Cognitive Science 7, 281 – 328. 1983.

Schank, R. C., *Dynamic Memory: A Theory of Reminding and Learning in Computers and People.* Cambridge: Cambridge University Press, 1982.

Schank, R. C., and D. B. Leake, Computer understanding and creativity. Information Processing 86, pp 335 – 341.

Shouksmith, G., *Intelligence, Creativity and Cognitive Style.* New York: Wiley-Interscience, 1970.

Simpson, R. L., A computer model of case-based reasoning in problem solving: An investigation in the domain of dispute mediation. Tech. Rep. GIT-ICS-95/18, Georgia Institute of Technology, Atlanta, GA, 1985.

ACTION SYSTEMS: PLANNING AND EXECUTION

Emilio Bizzi
Department of Brain and Cognitive Sciences
Massachusetts Institute of Technology
Cambridge, Massachusetts

It has been proposed that the central nervous system (CNS) generates movement as a shift of the limb's equilibrium posture. This equilibrium- point hypothesis has been corroborated by experimental results obtained in studies involving single- and multi-joint motions. Work on single-joint movements has, in fact, provided evidence for the spring-like behavior of muscles and has led to the suggestion that this property may play an important role in the control of posture and movement. These studies support the idea that posture control is achieved by the CNS through the choice of agonist-antagonist torque-angle curves that determine the equilibrium position for the limb and the stiffness about the joints. These results are also consistent with the idea that the CNS produces single- and multi-joint arm trajectories by generating a control signal that defines a series of equilibrium postures. These studies have achieved a unified description of posture and movement. The importance of the equilibrium- point hypothesis is that it drastically simplifies the computations the CNS presumably must make to accomplish multi-joint movements. Because the neuromuscular system is spring-like, at each instant the difference between the arm's actual position and the equilibrium position specified by the neural activity is sufficient to generate the torques required for movement. The complex problem of computing the torques at the joints (the so-called "inverse dynamic" problem) is avoided.

Up to this time, the experimental evidence for the equilibrium-point hypothesis rested upon data derived from psychophysical and behavioral experiments, and no experimental paradigm existed with which to test the implications of this hypothesis at the neurophysiological level. Recently, we have investigated the organization of limb motor space in the frog's spinal cord and found that microstimulation of the spinal cord's premotoneural network produces goal-directed leg movements that bring the leg to a particular position in the animal's motor space. These results provide the neurophysiological underpinning for the equilibrium-point hypothesis.

The forward modeling approach in speech production and limb control

Michael I. Jordan
Department of Brain and Cognitive Sciences
Massachusetts Institute of Technology
Cambridge, Massachusetts

The problem of learning to control an unknown environment is essentially that of finding an inverse of the state transition function of the environment. The forward modeling approach to motor learning (Jordan, 1990) is an indirect approach to finding such an inverse. The idea is to first learn a forward model of the state transition function (over an appropriate subdomain) and then to learn the identity mapping across the composition of the controller and the forward model. This process solves implicitly for an inverse mapping. The forward modeling approach is particularly appropriate for systems with excess degrees of freedom and it allows a priori knowledge to be incorporated during learning in the form of general constraints on the particular solution.

In this presentation I illustrate aspects of the forward modeling approach in the domains of speech production and limb control. The first example demonstrates that in a stochastic estimation problem the approach leads to biased inverse mappings that have smaller variance than the exact inverse mapping. This implies that over successive generations the targets that are passed from one learner to another tend to evolve to configurations of minimal variance. In the domain of speech production, this evolution corresponds to the claim of Stevens (1989) that sound inventories of languages tend to be composed of phonemes that have maximally stable articulatory configurations. I demonstrate this evolutionary process using a simulated vocal tract and the learning of a vowel system. The second example demonstrates that the forward modeling approach provides a technique for learning the virtual equilibrium trajectories associated with limb movements. I show that the use of virtual equilibrium trajectories as control signals leads to advantages associated with the stability of the learning process. Finally, I show that the forward modeling approach provides an account of a speed-accuracy tradeoff known as Fitt's Law.

References

Jordan, M.I. (1990). Motor learning and the degrees of freedom problem. In M. Jeannerod (Ed.), *Attention and Performance, XIII.* Hillsdale, NJ: Erlbaum.

Stevens, K.N. (1989). On the quantal nature of speech. *Journal of Phonetics, 17*, 3-45.

COGNITIVE ASPECTS OF LINGUISTIC THEORY

(Organizers: Howard Lasnik (Connecticut), David Pesetsky (MIT))

This symposium will explore cognitive issues arising from recent research in three areas of theoretical linguistics: syntax, semantics and phonology. Though the issues examined will lie at the frontiers of current work, no formal background in linguistics will be presupposed.

In his presentation, Prof. Howard Lasnik (University of Connecticut) will discuss the boundary between syntax and pragmatics as it is revealed through current studies of pronominal anaphora. Lasnik will show how considerations arising from the communicative function of language account for only the most global constraints on pronominal reference (.e.g the infelicity of He came in and John sat down, with he understood as John). By contrast, syntactic conditions are necessary to explain the intricacies of pronoun distribution. The study of such conditions shows that linguistic competence includes a distinct syntactic component with rich internal structure.

Prof. James Higginbotham (MIT) will address cognitive issues in semantics that stem from studies of verbal aspect. These studies have revealed a variety of scenarios that can be associated with simple sentences, e.g. the two senses of John raised his hand, one of which is like the sense of I raised John's hand. He will show how such associations are explained via a concept of logical form that recognizes sentences as classifying events. This theory can not only explain the logical links among sentences but can also account for the links between sentences and the perceptual situations that justify our classifications.

Prof. John McCarthy (University of Massachusetts) will focus on modern theories of non-linear phonology and what they reveal about the mind. Non-linear phonology differs from earlier theories in its focus on long-distance relations between phonological units that go beyond simple concatenation. Non-linear phonology posits a hierarchical arrangement of these units, whose structural properties are crucial to the explanation of phonological processes. McCarthy's presentation will examine some of its consequences of this approach to phonology.

Computational Models of Category Learning

DORRIT BILLMAN (BILLMAN@PRAVDA.GATECH.EDU)
School of Psychology, Georgia Institute of Technology, Atlanta, Georgia 30332

DOUGLAS FISHER (DFISHER@VUSE.VANDERBILT.EDU)
Department of Computer Science, Vanderbilt University, Nashville, TN 37235

MARK GLUCK (GLUCK@PSYCH.STANFORD.EDU)
Department of Psychology, Stanford University, Stanford, CA 94305

PAT LANGLEY (LANGLEY@PTOLEMY.ARC.NASA.GOV)
AI Research Branch, MS 244-17, Nasa Ames Research Center, Moffett Field, CA 94035

MICHAEL PAZZANI (PAZZANI@ICS.UCI.EDU)
Department of Information and Computer Science, University of California, Irvine, CA 92717

1 Introduction

Computational models have played an important role in cognitive science, giving researchers a precise, unambiguous language in which to express theories of cognition. Early work in cognitive simulation focused on the collection of verbal protocols for a small number of subjects, the detailed analysis of those protocols, and the creation of programs that simulated their behavior in considerable detail. This approach, exemplified by the work of Newell and Simon (1972), has led to important insights about human cognition, particularly in the area of problem solving.

Another early approach, exemplified by Feigenbaum's (1963) work on EPAM, focused on robust empirical generalizations, showing how such phenomena arose as emergent properties of a computational model. Although originally less common that the former approach, in recent years this research paradigm has been gaining ascendancy. In this paper, we consider theories of category learning that have taken this form. First, we summarize several memory, reasoning, and learning phenomena that models of category learning must explain. Next, we give brief overviews of three category learning models and indicate how these models account for some of the empirical findings. Finally, we discuss some open issues and consider promising directions for future research.

2 Empirical Generalizations about Categorization

Theories attempt to explain empirical phenomena, so we begin by reviewing some generalizations that have emerged from the experimental study of human categorization. We have not attempted to be exhaustive, but we believe the statements that follow provide important constraints on theories of category learning.

1. People are able to represent, access, and acquire concepts that involve logical 'rules' (Bourne, 1966), but they can also handle 'fuzzy' categories for which no 'logical' rules exist (Smith & Medin, 1981; Barsalou, 1985). For example, one might define birds as flying, beaked animals, but some birds cannot fly and some have bills instead of beaks.
2. Categories are influenced by the informational structure of the environment (Rosch, Mervis, Johnson, Gray, & Boyes-Braem, 1976), in that different experiences lead to different concepts. However, they are also influenced by the goals of the perceiver (Barsalou, 1983a, 1983b) and by intuitive beliefs and theories of the world (Chapman & Chapman, 1969; for discussion, Murphy & Medin, 1985).
3. People can detect and exploit correlations among features (Medin, Altom, Edelson, & Freko), as well as information about the

frequency of categories (Medin & Edelson, 1988; Gluck & Bower, 1988a). For instance, feathers are positively correlated with beaks, whereas fur tends to co-occur with teeth. Further, correlations are learned more easily when part of a coherent set of relations (Billman, 1989; Billman & Jeong, 1989).

4. People divide categories into subcategories, with some levels being more 'natural' than others. These 'basic' categories tend to occur at intermediate levels of abstraction. For instance, the category 'bird' is more basic than the categories 'animal' or 'robin'. Research with natural and artificial categories indicates that such concepts are learned earlier developmentally and more rapidly in experiments (Corter, Gluck, & Bower, 1988; Rosch et al., 1976). Objects are often identified at the basic level most rapidly, though this interacts with the typicality of the object (Murphy & Brownell, 1985).
5. Some instances are more 'typical' of a category than others (Rosch & Mervis, 1975). These are named more frequently and accessed more rapidly than less typical ones. For instance, robins are more typical birds than are penguins, and pictures of the former are recognized as birds more quickly than the latter. However, typicality varies across individuals and contexts (Barsalou, 1985, 1987, 1989). Moreover, there is dissociation between membership and typicality in some categories (Armstrong, Gleitman, & Gleitman, 1983) but not in others (Fehr & Russell, 1984; Hampton, 1987).
6. Just as category learning is influenced by prior beliefs, recognition of items as category members includes a top-down aspect: entities are categorized faster in expected contexts than in unexpected contexts (Palmer, 1975). For example, a drawing of a loaf of bread is recognized more rapidly when located in a kitchen than in a street scene.
7. People can represent, access, and acquire categories that involve structural and conceptual relations between components (Barsalou, in press; Fodor & Pylyshyn, 1988). For instance, the relative locations of the eyes and nose are essential aspects of the concept 'face'.
8. People use categories to guide inference as well as classification and recognition. Inferences about new instances are guided by category membership. For example, given that a novel entity can be identified as a bird, one can infer that it probably hatched from an egg.
9. Categories are used in inferences about new properties. Generalization of new properties to individual instances and to sets of instance is guided by category membership (Gelman & Markman, 1987; Nisbett, Ross, Jepson, & Kunda). Indeed, under some conditions, people generalize properties of a single exemplar to an entire category (Osherson, Smith, Wilkie, Lopes, & Shafir, in press; Macario, Shipley, & Billman in press).

In sum, theories should eventually account for category learning, for classification, and for use of categories in inference. Both characteristics of the input (e.g., correlations among attributes, feature frequency) and background knowledge influence all these processes. In the following sections, we provide overviews of three computational models of category formation. In each case, we describe the model's representation of conceptual knowledge, the manner in which that knowledge is used, and the learning mechanisms through which it is acquired. We also examine each model's ability to explain some subset of the empirical generalization listed above.

3 Configural-Cue Adaptive Networks

Gluck and Bower (1988a, 1988b) describe an adaptive network model of human learning that extends Rescorla and Wagner's (1972) theory of classical conditioning to human classification learning. The model represents knowledge as a set of one-layer classifiers, one for each category. Each network has a set of input that correspond to features that may occur in an experience, a set of weighted links, and a single output node. Given a new experience, one outputs a classification probability by adding the weights on matched input features. Learning involves changing the weights on links using Widrow and Hoff's (1960) least mean squares method. Briefly, this alters weights so as to decrease the difference between the actual and desired score for each classifier.

This adaptive network model has accurately accounted for human behavior in experiments on probabilistic classification learning with multiple cues, but it can only learn 'linearly separable' categories, that is, ones that can be separated by a hyperplane through the space of instances. However, one can extend the model to non-linearly separable categories by letting conjunctions of elementary stimulus features serve as 'higher-order' features of a stimulus pattern. Thus, given the presentation of an experimental pattern consisting of elementary features BCD, one assumes that this is reflected not only through activation of input nodes for the single elements B, C, and D, but also through activation of the pair-wise conjuncts BC, BD, and CD. In this approach, a domain involving N elementary features would be represented using $N^2 + N$ input features for each category, but the categorization and learning mechanisms would remain unchanged.

Gluck, Bower, and Hee (1989) have shown how this extended model accounts for several aspects of complex category learning by humans. Moreover, Corter, Gluck, and Bower (1988) have applied the configural-cue approach to model basic-level effects in hierarchically organized categories. For instance, when the network model is trained on three levels of an artificial domain involving hierarchical category structure, high levels of activation are reached sooner for the output nodes corresponding to the intermediate (basic) level categories. The predicted learning curves closely resemble the observed curves for humans, and the model also correctly predicts a shift in relative difficulty between the subordinate and superordinate levels in different experimental conditions.

A key property of the configural-cue model is that it embodies, implicitly, an approximate exponential decay relationship between stimulus similarity and psychological distance, a relationship with considerable independent support in studies of stimulus generalization (Shepard, 1958) and categorization (Nosofsky, 1984). This effect can be seen by noting how the number of overlapping active nodes (similarity) changes as a function of the number of overlapping component cues (distance). If two triplet patterns share one feature (ABC, XYC), they will have only one active node in common and five nodes nonoverlapping; if they share two features (ABC, XBC), they will have three active nodes in common (two component cues and one configural-cue node) and three nonoverlapping nodes. In fact, the network model can be viewed as an extension of Shepard's (1987) theory of stimulus generalization to classification learning.

In addition to basic-level effects, the configural-cue model is consistent with many of the phenomena from Section 2. It can certainly acquire non-logical categories, and its use of pair-wise features lets it capture correlations. The concepts learned by the model are certainly a function of the environment it experiences, though this is represented only in different weights. In summary, combining the adaptive network model with a representation of stimuli that includes pair-wise configurations of features lets one account for a wider range of learning results from both the animal and human learning literatures.

The configural-cue model has several obvious limitations, including the rapid growth of input nodes with increasing pattern size. In addition, the approach can only make predictions about prelabeled classes, and it cannot handle structural representations of knowledge. Nevertheless, the model is theoretically parsimonious, accounts for a wide range of phenomena, and uses assumptions for which independent evidence already exists. Furthermore, its successes are instructive in identifying empirical phenomena that can be explained as emergent from the same elementary, associative processes found in lower species.

4 Representation Change in Concept Formation

Although some concept-learning tasks involve supervision, people also acquire conceptual knowledge without explicit supervision. The research described in this section takes the primary task of category learning, particularly unsupervised learning, to be recovery of the correlational structure of input. This produces coherent categories useful in prediction and inference. Models developed in this framework directly represent correlational structure as probabilistic patterns or rules. Further, these models assume category formation is intimately linked to representation change, and specifically to change in the attributes and features used to represent input. Representation change is important, but little studied from the perspective of concept learning. We distinguish between two types of representation change.

'Weak' representation change involves a change in attention to or significance of a property. In contrast, 'strong' representation change involves the introduction of new properties (a 'limiting case' of increase in attention).

CARI (Billman & Heit, 1988) and descendant models (Chalnick & Billman, 1988) investigated procedures for attention change in unsupervised learning. In particular, increasing attention is directed to attributes that prove predictive in some relation; this facilitates the discovery of other related predictive patterns when input exhibits any of a broad class of coherent structures (psychological motivation from Billman, 1989). More recently, Billman and Martin have explored a context-sensitive form of attentional learning. The idea is derived from the philosopher Goodman's (1955, 1984) concept of overhypotheses (see also Russell, 1986, on well-defined categories). Categories of a particular type (jewels, animals, ethnic groups) are homogeneous – or predictable – with respect to some properties (crystal structure, diet, language) but not others (weight, age, first name). Members of one category share values of the homogeneous attributes and contrast with other categories on these attributes. This information can be used to make inferences about novel categories and attribute values. Seeing one instance of a new kind of jewel, one would generalize its crystal structure but not its weight to other category members. Further, people do seem to make use of this information (Nisbett, Krantz, Jepson, & Kunda, 1983; Shipley 1989).

In related work, Billman and Martin have investigated two types of 'strong' representation change. One concerns the formation of new attributes from previously uncoordinated features; thus, someone may come to see legs, fins, and wings as values of a new attribute 'limbs'. The second involves the formation of new features from conjoining old features or attribute values. Martin and Billman (in press; Martin, 1989) describe CORA, a computational model of this latter process, which we now describe in some detail.

CORA represents conceptual knowledge in terms of probabilistic inference rules. For instance, it might have the rule IF X HAS WINGS AND X HAS FEATHERS, THEN INFER THAT X FLIES WITH PROBABILITY 0.95. In this case, a conjunction of two features predicts the presence of a third, but simpler and more complex rules are possible. CORA does not represent concepts as separate knowledge structures; rather, it organizes knowledge an interconnected network that directly represents conditional probabilities between features and conjunctions of features. The system also contains knowledge about features which are mutually exclusive, that is, which are alternative values of the same attribute.

Given a new experience with the values of some attributes omitted, CORA uses its inference rules to predict the missing values. For each attribute, it applies all rules that infer a value for that attribute and whose conditions match against the new experience. The system then estimates the overall probability for each possible value as the geometric average of the individual predicted probabilities. Finally, CORA predicts the value of the attribute that has the highest overall probability.

Billman and Martin's system begins with simple inference rules in which the condition sides contain only one feature, and in which all rules have the same associated probability. However, CORA augments this knowledge using two learning mechanisms. First, when the system is given a new experience, it iterates through each observed value, updating the conditional probability of each rule that infers that value. After this, it checks to determine whether the updated rules – in combination – actually predict the observed value over the alternatives. If they do not, this suggests that there are interactions among features which are not being taken into account, and CORA creates a new inference rule whose condition side is the conjunction of two existing rules. The component rules are those which have been most frequently associated with errors in the past. In this way, the system begins with simple inference rules and incrementally constructs more complex ones in which higher-order features are present.

The basic approach shares several characteristics with Gluck's configural-cue model. Both systems represent conjunctions of features, associate weights with these higher-order terms, and combine the relevant 'rules' to make predictions. However, the configural-cue model assumes that all pairwise conjunctions are present from the outset, whereas CORA begins with single features and constructs higher-order terms as necessary. Thus, Gluck's learning method searches only the space

of weights on prespecified links, whereas Martin and Billman's system carries out a general-to-specific search through the space of feature combinations. Another difference is that CORA is unsupervised, in that it requires no labeled training instances and predicts the value of any missing attribute.

The CORA model is consistent with a number of the phenomena given in Section 2. The system can handle both logical 'rules' and more 'fuzzy' representations of knowledge, and it constructs different inference rules depending on the informational structure of the environment. CORA predicts that some experiences are more typical than others, but it makes no assumptions about reaction times. The model can certainly exploit correlations among features, and it emphasizes the use of conceptual knowledge for prediction. However, it provides no account of basic-level effects, and it fails to address the structural nature of some knowledge. Nevertheless, the approach provides a promising account of the representation, use, and acquisition of conceptual knowledge, and future work may address these issues.

5 Hierarchical Concept Formation

Many models of concept formation construct classification hierarchies over environmental observations. Perhaps the earliest such system was Feigenbaum's (1963) EPAM, which incrementally formed discrimination networks and used them to classify new observations. Lebowitz's (1982) UNIMEM and Kolodner's (1983) CYRUS built more sophisticated, redundant classification structures, but they sifted observations down alternative paths in a hierarchy much like EPAM.

Here we will focus on Fisher's (1987) COBWEB, which was intended to synthesize ideas from these earlier systems and thus will serve briefly to highlight characteristics of the general hierarchical approach.[1] In particular, the system uses a probabilistic representation of concepts (Smith & Medin, 1981). Each concept specifies a set of attributes and their possible values, along with the conditional probability of that value given the concept. COBWEB also stores the overall probability of occurrence for each concept. Moreover, concepts are organized into a concept hierarchy that is partially ordered according to specificity, with more abstract categories at higher levels, more specific ones at lower levels, and specific observations as terminal nodes.

Like its predecessors, COBWEB sorts new observations downward through its hierarchy, selecting the best branch at each level. In making this decision, it finds the 'best match' according to *category utility*, an evaluation function proposed by Gluck and Corter (1985) to account for basic levels observed in experimental studies of human categorization. They arrived at this function through a rational analysis (Anderson, in press) of the categorization task, in which basic-level concepts are preferred because they facilitate more accurate predictions about their members.

Like Martin and Billman's model, COBWEB acquires concepts incrementally, during the process of categorization. As it sorts experiences down the hierarchy, the system alters the probabilities stored with each concept and its associated values. In some cases, COBWEB also changes the structure of its hierarchy. If an experience differs sufficiently from the children of a concept, the system stores it as a new child at that level. In other cases, COBWEB finds that merging or splitting existing concepts at a given level will improve the match score. The structure of the resulting hierarchy is a function not only of the experiences given to the system, but of the order in which they are presented.

Note that COBWEB differs from CORA in that it does not explicitly compute correlations among features. Category utility sums a function of individual feature probabilities, so that each node in memory effectively represents a category as an independent cue model (Smith & Medin, 1981). Fisher and Langley (in press) have shown that favoring the creation of categories that maximize this function of individual features will *tend* to reward categories that capture correlations. However, leaves of the concept hierarchy correspond to specific observations (cases, exemplars), giving representational power equivalent to that of exemplar (Smith & Medin, 1981) or relational cue (Medin, 1983) representations. In addition, hierarchical systems offer a natural representation of default and exceptional properties: one assumes the most likely attribute value for a given category, unless and the observed features warrant deeper classification.

[1] We direct readers Gennari, Langley, and Fisher (1989) for a review of hierarchical approaches to concept formation.

COBWEB and most other hierarchical systems assume that observations are represented as nominal attribute-value pairs. However, recent work has extended the basic framework to handle real-valued attributes (Gennari, Langley, & Fisher, 1989) and structured descriptions (Thompson & Langley, 1989), in which acquired concepts are defined in terms of other acquired concepts and relations among them. Moreover, some researchers have adapted such relational representations for use in concept formation over traces of problem-solving experiences (e.g., Yang, Yoo, & Fisher, in press). In this approach, previous experience is organized in memory and accessed for use in guiding behavior on novel problems. Such models may explain the origin functional categories, as well as predicting psychological phenomena in problem-solving domains that are analogous to those found in simple categorization tasks.

Despite its initial motivation in terms of computational efficacy, COBWEB also accounts for a variety of psychological phenomena (Fisher & Langley, in press). Its account of basic-level effects follows from its use of category utility, but with assumptions about retrieval rates, accounts of typicality effects and fan effects (Anderson, 1976) also emerge. Like the other models we have examined, COBWEB can represent and acquire nonlogical categories, but it can handle logical rules as a special case. The system takes advantages of correlations among features, though it represents them only indirectly, and Fisher (1987) has extensively tested its predictive ability. Extensions to the framework show promise for structural knowledge, and the model even makes certain predictions about interactions between typicality and basic levels (Fisher, 1988). Thus, hierarchical methods for concept formation provide another promising framework for explaining the nature of human categorization.

6 Conclusions

We have presented three models that account for some empirical findings in category formation. No one model claims to account for all of the complexity of category formation. In some cases, techniques used in one model can address shortcomings of other models. For example, CORA suggests a way of going beyond pairwise conjunctions in the configural cue model while avoiding the enumeration of all possibilities. In some cases, the models provide contrasting accounts for the same phenomenon. For example, both COBWEB and the configural-cue model propose different mechanisms to account for basic-levels effects.

Some phenomena are not addressed by any of the models. In particular, none provide an explicit account for the priming effect (e.g., Palmer, 1975) of expected contexts, though the use of conditional probabilities in CORA and COBWEB suggest possible extensions. All of the current models emphasize the role of the informational structure of the environment, ignoring the role played by the learner's goals and theories of the world in category formation.

We have focused on demonstrating that empirical generalizations of human learning such as basic-level and typicality effects can arise as emergent properties of computational models. It is equally important that these computational models make predictions and be used to formulate hypotheses about human behavior that can be tested empirically. Furthermore, the precision demanded by computational models often raises issues that might have otherwise been overlooked. Thus, we feel that experimentation and computational modeling play essential complementary roles in understanding cognition.

Acknowledgements

We would like to thank Larry Barsalou for his advice on the complex topic of categorization and Joel Martin for comments on an earlier draft.

References

Anderson, J. R. (1976). *Language, memory, and thought*. Hillsdale, NJ: Lawrence Erlbaum.

Anderson, J. R. (in press). The place of cognitive architectures in a rational analysis. In K. Van Lehn (Ed.), *Architectures for intelligence*. Hillsdale, NJ: Lawrence Erlbaum.

Armstrong, S. L., Gleitman, L. R., & Gleitman, H. (1983). On what some concepts might not be. *Cognition*, *13*, 263–308.

Barsalou, L. W. (1983a). Ad hoc categories. *Memory & Cognition*, *11*, 211–227.

Barsalou, L. W. (1983b). Context-independent and context-dependent information in concepts. *Memory & Cognition*, *10*, 83–93.

Barsalou, L. W. (1985). Ideals, central tendency, and frequency of instantiation as determinants of graded structure in categories. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 11*, 629–654.

Barsalou, L. W. (in press). Frames, concepts, and conceptual fields. In A. Lehrer & E. Kittay (Eds.), *Semantic fields and other alternatives*.

Billman, D. (1989). Systems of correlations in rule and category learning: Use of structured input in learning syntactic categories. *Language and Cognitive Processes, 4*, 127–155.

Billman, D. O., & Heit, E. (1988). Observational learning from internal feedback: A simulation of an adaptive learning method. *Cognitive Science, 12*, 587–626.

Billman, D., & Jeong, A. (1989). Systematic correlations facilitate learning component rules in spontaneous category formation. Paper presented at the 30th Annual Meeting of the Psychonomic Society, November, 1989, Atlanta, GA.

Bourne, L. E. (1966). *Human conceptual behavior*. Boston: Allyn and Bacon.

Chapman, L. J., & Chapman, J. P. (1969). Illusory correlation as an obstacle to the use of valid psychodynamic signs. *Journal of Abnormal Psychology, 74*, 272–280.

Corter, J. E., Gluck, M. A., & Bower, G. H. (1988). Basic levels in hierarchically structured categories. *Proceedings of the Tenth Annual Conference of the Cognitive Science Society*. Montreal, Canada: Lawrence Erlbaum.

Fehr, B., & Russell, J. A. (1984). Concept of emotion viewed from a prototype perspective. *Journal of Experimental Psychology: General, 113*, 464–486.

Feigenbaum, E. A. (1963). The simulation of verbal learning behavior. In E. A. Feigenbaum & J. Feldman (Eds.), *Computers and thought*. New York: McGraw-Hill.

Fisher, D. H. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine Learning, 2*, 139–172.

Fisher, D. (1988). A computational account of basic level and typicality effects. *Proceedings of the Seventh National Conference on Artificial Intelligence* (pp. 233–238). St. Paul, MN: Morgan Kaufmann.

Fisher, D. & Langley, P. (in press). The structure and formation of natural categories. In G. Bower (Ed.), *The psychology of learning and motivation* (Vol. 26). New York: Academic Press.

Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition, 28*, 3–71.

Gennari, J., Langley, P., & Fisher, D. (1989). Models of incremental concept formation. *Artificial Intelligence, 40*, 11–62.

Gluck, M. A., & Bower, G. H. (1988a). Evaluating an adaptive network model of human learning. *Journal of Memory and Language, 27*, 166–195.

Gluck, M. A., & Bower, G. H. (1988b). From conditioning to category learning: An adaptive network model. *Journal of Experimental Psychology: General, 117*, 225–244.

Gluck, M. A., & Corter, J. E. (1985). Information, uncertainty, and the utility of categories. *Proceedings of the Seventh Annual Conference of the Cognitive Science Society* (pp. 283–287). Irvine, CA: Lawrence Erlbaum.

Gluck, M. A., Bower, G. H., & Hee, M. R. (1989). A configural-cue network model of animal and human associative learning. *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*. Ann Arbor, MI: Lawrence Erlbaum.

Goodman, N. (1955/1984). *Fact, fiction, and forecast*. New York: Bobbs-Merrill.

Hampton, J. A. (1987). Overextension of conjunctive concepts: Evidence for a unitary model of concept typicality and class inclusion. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 14*, 12–32.

Kolodner, J. L. (1983). Reconstructive memory: A computer model. *Cognitive Science*, 7, 281–328.

Lebowitz, M. (1982). Correcting erroneous generalizations. *Cognition and Brain Theory, 5*, 367–381.

Macario, J., Shipley, E., & Billman, D. (in press). Induction from a single instance: Formation

of a novel category. *Journal of Experimental Child Psychology*.

Martin, J. D. (1989). Focusing attention for observational learning: The importance of context. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*. Detroit, MI: Morgan Kaufmann.

Medin, D. (1983). Structural principles of categorization. In T. Tighe & B. Shepp (Eds.), *Perception, cognition, and development*. Hillsdale, NJ: Lawrence Erlbaum.

Medin, D. L., Altom, M. W., Edelson, S. M., & Freko, D. (1982). Correlated symptoms and simulated medical classification. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *8*, 37–50.

Murphy, G. L., & Brownell, H. H (1985). Category differentiation in object recognition: Typicality constraints on the basic category advantage. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *11*, 70–84.

Medin, D. L., & Edelson, S. M. (1988). Problem structure and the use of base-rate information from experience. *Journal of Experimental Psychology: General*, *117*, 68–85.

Murphy, G. L., & Medin, D. L. (1985). The role of theories in conceptual coherence. *Psychological Review*, *92*, 289–316.

Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice-Hall.

Nisbett, R. E., Krantz, D. H., Jepson, C., & Kunda, Z. (1983). The use of statistical heuristics in everyday deductive reasoning. *Psychological Review*, *90*, 339–363.

Nosofsky, R. M. (1984). Choice, similarity, and the context theory of classification. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *10*, 104–114.

Osherson, D. N., Smith, E. E., Wilkie, O., Lopez, A., & Shafir, E. (in press). Category based induction. *Psychological Review*.

Palmer, S. E. (1975). The effects of contextual scenes on the identification of objects. *Memory & Cognition*, *3*, 519–526.

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and non-reinforcement. In A. H. Black, & W. F. Prokasy (Eds.), *Classical conditioning II: Current research and theory*. New York: Appleton-Century-Crofts.

Rosch, E. (1978). Principles of categorization. In E. Rosch and B. Lloyd (Eds.), *Cognition and categorization*. Hillsdale, NJ: Lawrence Erlbaum.

Rosch, E., & Mervis, C. (1975). Family resemblances: studies in the internal structure of categories. *Cognitive Psychology*, *7*, 573–605.

Rosch, E. H., Mervis, C. B., Gray, W. D., Johnson, D. M., & Boyes-Braem, P. (1976). Basic objects in natural categories. *Cognitive Psychology*, *8*, 382–439.

Russell, S. J. (1986). Preliminary steps toward the automatization of induction. *Proceedings of the Fifth National Conference on Artificial Intelligence*. Philadelphia, PA: Morgan Kaufman.

Shepard, R. N. (1958). Stimulus and response generalization: Deduction of the generalization gradient from a trace model. *Psychological Review*, *65*, 242–256.

Shepard, R. N. (1987). Towards a universal law of generalization for psychological science. *Science*, *237*, 1317–1323.

Shipley, E. F. (1990). *Hierarchies and induction.* Unpublished manuscript, Department of Psychology, University of Pennsylvania, Philadelphia, PA.

Smith, E. E., & Medin, D. L. (1981). *Categories and concepts*. Cambridge, MA: Harvard University Press.

Thompson, K., & Langley, P. (1989). Incremental concept formation with composite objects. *Proceedings of the Sixth International Workshop on Machine Learning* (pp. 371–374). Ithaca, NY: Morgan Kaufmann.

Widrow, B., & Hoff, M. E. (1960). Adaptive switching circuits. *Institute of Radio Engineers, Western Electronic Show and Convention, Convention Record*, *4*, 96–194.

Yoo, H., Yang, H., & Fisher, D. (in press). Concept formation over problem-solving experience: Issues of induction and utility. In D. Fisher & M. Pazzani (Eds.), *Computational approaches to concept formation*. San Mateo, CA: Morgan Kaufmann.

Designing an Integrated Architecture: The PRODIGY View *

Jaime G. Carbonell, Yolanda Gil, Robert Joseph,
Craig A. Knoblock, Steve Minton†, and Manuela M. Veloso

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

† Sterling Federal Systems
NASA Ames Research Center
AI Research Branch, Mail Stop: 244-17
Moffett Field, CA 94035

ABSTRACT

Artificial intelligence has progressed to the point where multiple cognitive capabilities are being integrated into computational architectures, such as SOAR, PRODIGY, THEO, and ICARUS. This paper reports on the PRODIGY architecture, describing its planning and problem solving capabilities and touching upon its multiple learning methods. Learning in PRODIGY occurs at all decision points and integration in PRODIGY is at the knowledge level; the learning and reasoning modules produce mutually interpretable knowledge structures. Issues in architectural design are discussed, providing a context to examine the underlying tenets of the PRODIGY architecture.

1 Introduction

A common dream for many AI researches, present authors included, is the construction of a general purpose learning and reasoning system that given basic axiomatic knowledge of a domain is capable of becoming an expert problem solver.

Our machine learning approach, implemented in PRODIGY [Carbonell *et al.*, 1990], starts with a general problem-solving engine based on a possibly incomplete domain theory. The problem solver improves its performance through experience by refining the initial domain knowledge and learning knowledge to control the search process. The paper is divided into two parts. The first part describes the basic architecture, including the problem solver and the various learning modules. The second part discusses the design issues in building an integrated architecture.

---

*This research was sponsored in part by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 4976, Amendment 20, under contract number F33615-87-C-1499, monitored by the Air Force Avionics Laboratory, in part by the Office of Naval Research under contracts N00014-84-K-0345 (N91) and N00014-86-K-0678-N123, in part by NASA under contract NCC 2-463, in part by the Army Research Institute under contract MDA903-85-C-0324, and in part by small contributions from private institutions. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of DARPA, ONR, NASA, ARI, or the US Government. The fourth author was supported by an Air Force Laboratory Graduate Fellowship, and the third author was supported by an AT&T Bell Labs Ph.D. Scholarship.

2 The PRODIGY Architecture

2.1 The Problem Solver

PRODIGY's basic reasoning engine is a general-purpose problem solver and planner [Veloso, 1989, Minton *et al.*, 1989] that searches for sequences of operators (i.e., plans) to accomplish a set of goals from a specified initial state description. Search in PRODIGY is guided by a set of *control rules* that apply at each decision point. Search control rules may be general or domain specific, hand-coded or automatically acquired, and may consist of heuristic preferences or definitive selections. In the absence of any search control, PRODIGY defaults to depth-first means-ends analysis. But, with appropriate search control rules it can emulate other search disciplines, including breath-first search, depth-first iterative-deepening, best-first search, and knowledge-based plan instantiation.

2.1.1 Domains of application

PRODIGY has been applied to many different domains: robotic path planning, the blocks-world, an augmented version of the STRIPS domain, matrix algebra manipulation, discrete machine-shop planning and scheduling, computer configuration, logistics planning, and several others. In order to solve problems in a particular domain, PRODIGY must first be given a specification of that domain, consisting of a set of operators and inferences rules.

2.1.2 Knowledge representation

Each operator has a precondition expression that must be satisfied before the operator can be applied, and a list of effects that describe how the application of the operator changes the world. Precondition expressions are well-formed formulas in PDL, a form of predicate logic encompassing negation, conjunction, disjunction, and existential and universal quantification. The effects are atomic formulas that describe the conditions that are added or deleted from the current state when the operator is applied. Operators may also contain conditional effects, which represent changes to the world that are dependent on the state in which the operator is applied.

2.2 Problem definition

A problem consists of an initial state and a goal expression. To solve a problem, PRODIGY must find a sequence of operators that, if applied to the initial state, produces a final state satisfying the goal expression. The search tree initially starts out as a single node containing the initial state and goal expression. The tree is expanded by repeating the following two steps:

1. **Decision phase:** There are four types of decisions that PRODIGY makes during problem solving. First, it must decide what node in the search tree to expand next, defaulting to a depth-first expansion. Each node consists of a set of goals and a state describing the world. After a node has been selected, one of the node's goals must be

selected, and then an operator relevant to this goal must be chosen. Finally, a set of bindings for the parameters of that operator must be decided upon.

2. **Expansion phase:** If the instantiated operator's preconditions are satisfied, the operator is applied. Otherwise, PRODIGY subgoals on the unmatched preconditions. In either case, a new node is created with updated information about the state or the subgoals.

The search terminates after creating a node whose state satisfies the top-level goal expression.

2.3 Control Rules

As PRODIGY attempts to solve a problem, it must make decisions about which node to expand, which goal to work on, which operator to apply, and which objects to use. These decisions can be influenced by control rules for the following purposes:

1. **To increase the efficiency of the problem solver's search.** Control rules guide the problem solver down the correct path so that solutions are found faster.
2. **To improve the quality of the solutions that are found.** There is usually more than one solution to a problem, but only the first one that is found will be returned. By directing the problem solver's attention along a particular path, control rules can express preferences for solutions that are qualitatively better (e.g., more reliable, less costly to execute, etc.).
3. **To direct the problem solver along paths that it would not explore otherwise.** As with most planners, for efficiency PRODIGY normally explores only a small portion of the complete search space. However, control rules can be used to force PRODIGY to explore a path that would not be expanded by the default search.

PRODIGY's reliance on explicit control rules, which can be learned for specific domains, distinguishes it from most domain independent problem solvers. Instead of using a least-commitment search strategy, for example, PRODIGY expects that any important decisions will be guided by the presence of appropriate control knowledge. If no control rules are relevant to a decision, then PRODIGY makes a quick, arbitrary choice. If in fact the wrong choice is made, and costly backtracking proves necessary, an attempt will be made to learn the control knowledge that must be missing. The rationale for PRODIGY's *casual commitment* strategy is that for any decision with significant ramifications, control rules should be present; if they are not, the problem solver should not attempt to be clever without knowledge, rather, the cleverness should come about as a result of learning. Thus, our emphasis is on an elegant and simple problem solving architecture which can produce sophisticated behavior by learning the appropriate, domain-specific control knowledge.

Control rules can be employed to guide the four decisions described in Section 2.2. Each control rule has a left-hand side condition testing applicability and a right-hand side indicating whether to SELECT, REJECT, or PREFER a particular candidate. To make a control decision, given a default set of candidates (nodes, goals, operators, or bindings, depending on the decision), PRODIGY first applies the applicable selection rules to select a subset of the

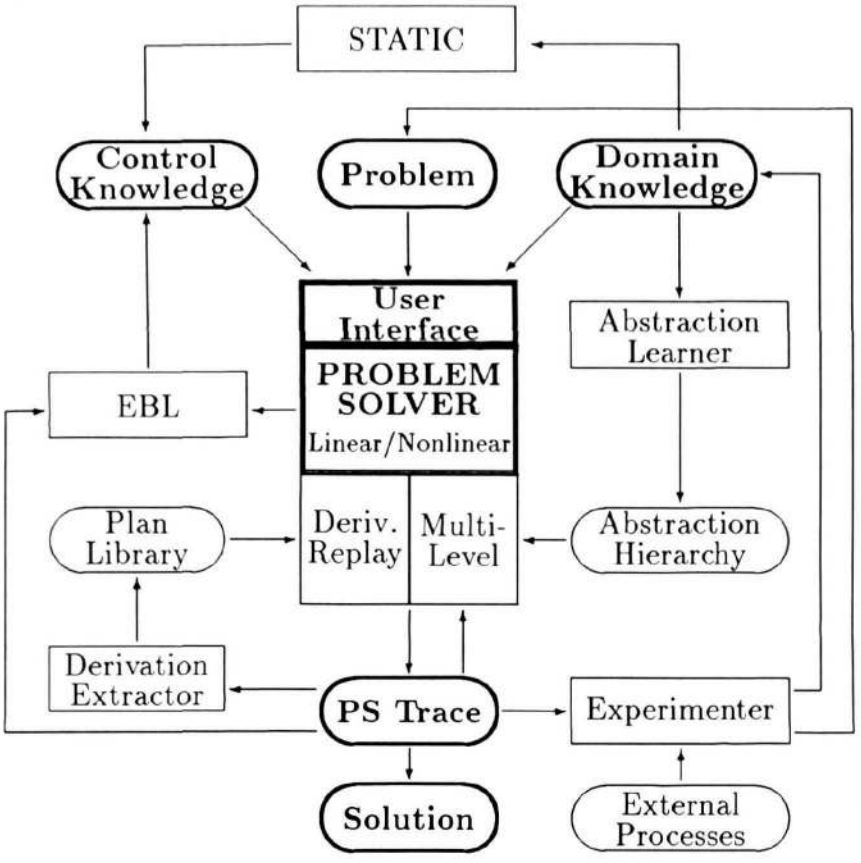


Figure 1: The PRODIGY Architecture

candidates. (If no selection rules are applicable, all the candidates are included.) Next rejection rules further filter this set by explicit elimination of particular remaining candidates, and finally preference rules are used to find the most preferred alternative. If backtracking is necessary, the next most preferred candidate is attempted, and so on, until a global solution is found, or until all selected and non-rejected candidates are exhausted.

2.4 The Learning Modules

PRODIGY's general problem solver is combined with several learning modules. The PRODIGY architecture was designed both as a unified testbed for different learning methods and as a general architecture to solve interesting problems in complex task domains. Let us now focus on the architecture itself, as diagrammed in Figure 1.

The problem solver produces a complete search tree, encapsulating all decisions – right ones and wrong ones – as well as the final solution. This information is used by each learning component in different ways. In addition to the central problem solver, PRODIGY has the following learning components:

APPRENTICE: A user interface that can participate in an apprentice-like dialogue [Joseph, 1989], enabling the user to evaluate and guide the system's problem solving and learning. The interface is graphic-based and tied directly to the problem solver, so

that it can both acquire domain knowledge or accept advice as it is solving a problem.

EBL: An explanation-based learning facility [Minton, 1988] for acquiring control rules from a problem-solving trace. Explanations are constructed from an axiomatized theory describing both the domain and relevant aspects of the problem solver's architecture. Then the resulting descriptions are expressed in control rule form, and control rules whose utility in search reduction outweighs their application overhead are retained.

STATIC: A method for learning control rules by analyzing PRODIGY's domain descriptions prior to problem solving. The STATIC program [Etzioni, 1990] produces control rules without utilizing any training examples. STATIC matches EBL's performance on some domains and exhibits one to two orders of magnitude faster learning rate. However, not all problem spaces permit purely static learning, requiring EBL to learn control rules dynamically.

ANALOGY: A derivational analogy engine [Carbonell and Veloso, 1988, Veloso and Carbonell, 1989] that uses similar previously solved problems to solve new problems. The problem solver records the justifications for each decision during its search process. These justifications are then used to guide the reconstruction of the solution for subsequent problem solving situations where equivalent justifications hold true. Both analogy and EBL are independent mechanisms to acquire domain-specific control knowledge.

ALPINE: An abstraction learning and planning module [Knoblock, 1990]. The axiomatized domain knowledge is divided into multiple abstraction levels based on an analysis of the domain. Then, during problem solving, PRODIGY first finds a solution in an abstract space and then uses the abstract solution to guide the search for solutions in more detailed problem spaces. This method is orthogonal to analogy and EBL, in that both can apply at each level of abstraction.

EXPERIMENT: A learning-by-experimentation module for refining domain knowledge that is incompletely or incorrectly specified. Experimentation is triggered when plan execution monitoring detects a divergence between internal expectations and external observations. The main focus of experimentation is to refine the factual domain knowledge, rather than the control knowledge.

3 Issues

Many controversial issues arise in the design and construction of an integrated architecture. These issues are resolved differently in PRODIGY, SOAR [Rosenbloom *et al.*, 1990], THEO [Mitchell *et al.*, 1990], ICARUS [Langley *et al.*, 1989], and other attempts at integrating multiple aspects of cognition and perception. PRODIGY takes specific positions in these design dimensions and derivative architectural attributes, which we elucidate below to promote discussion.

3.1 Types of Reasoning

Cognitive behaviors range from the emotive and intuitional to the rational and deliberative. Within the latter category, one can distinguish real-time decision making and long-term planning. PRODIGY models only deliberative planning and problem-solving, albeit in resource-limited domains.

Within planning and problem-solving task domains, several reasoning paradigms have been proposed, including least-commitment planning, "reactive planning", and casual commitment planning. PRODIGY subscribes to the latter, as explained in Section 2.3, learning new control knowledge from correct and incorrect past commitments. "Reactive planning", an archetypical oxymoron, refers to conditioned-reflex and the total absence of any higher-level deliberation or knowledge representation. We believe that whereas there is a role for such behavior, it is patently absurd to negate the need for higher level cognition as sometimes advocated by the proponents of "reactive planning" [Brooks, 1986, Agre and Chapman, 1987].

3.2 Modular vs. Monolithic Architecture

We do not know whether the human mind compartmentalizes distinct cognitive abilities, or distinct methods of achieving similar ends (such as our multiple learning methods). However, modularization at this level is a sound engineering principle, and therefore we have adhered to it. Integration is brought about by sharing both a uniform knowledge representation, and a common problem-solving and planning engine. Other systems, such as SOAR, take on a monolithic structure. There is only one learning mechanism, chunking, which can never be turned off or even modulated. Which is better? Clearly, we believe the former to be superior – but only from engineering principles rather than psychological ones.

3.3 Scaling Up

Any integrated architecture must address increasingly large tasks, whether its objective is to model human cognition or to build useful knowledge-based systems for complex tasks. Scalability can be calibrated in multiple ways, but all relate to efficient behavior with increasing complexity, as measured by:

- Size of the domain: total number of objects, attributes, relations, operators, inference rules, etc.
- Size of the problem: number of steps in the solution plan, number of conjuncts in the goal expression, size of the visited search space, etc.
- Variety: number of qualitatively different actions and object types in the domain.
- Perplexity: average fan-out at every decision point in the search space (with and without learned control knowledge).

In PRODIGY we seek to achieve a reasonable measure of scalability in all these dimensions. The learning techniques strive to reduce the visited search space in future problems with respect to the virtual (complete) search space.

3.4 Psychological Validity vs Cognitive Engineering

As mentioned earlier, the PRODIGY project strives to produce a useful, scalable, and maintainable reasoning and learning architecture. Where this matches human cognition, it is so by accident, by the limited imagination of the PRODIGY designers, or perhaps because the human mind has indeed optimized such aspects of cognition. In all other aspects, the goal is at reengineering cognition the way it ought to be, in order to be most useful in problem solving, planning, and learning. Here we enumerate a few additional ways in which PRODIGY differs from human thought and from other cognitive architectures:

- PRODIGY forgets when it chooses to do so. For instance, a control rule whose testing and application overhead is greater than the search reduction benefits accrued over time may be discarded. All other systems retain all acquired knowledge. Minton [Minton, 1988] demonstrated that the effectiveness of explanation-based learning is improved by measuring the utility of acquired knowledge and retaining only those rules with positive utility.
- PRODIGY deliberates on any and all decisions: which goal to work on next, which operator to apply, what objects to apply the operator to, where to backtrack given local failure, whether to remember newly acquired knowledge, whether to refine an operator that makes inaccurate predictions, and so on. It can introspect fully into its decision cycle and thus modify it at will. This is not consistent with the human mind, yet it is an extremely useful faculty for rapid learning.
- PRODIGY's knowledge acquired in one module is open to inspection and interpretation by other modules. Abstracted operators can be used to plan, to drive EBL, to analogize with past memory, and so forth. The compartmentalization is at the level of learning methods, and the sharing is at the level of all knowledge acquired. This is an architectural property different from those of other integrated architectures.

Acknowledgements

The authors gratefully acknowledge the contributions of the other members of the PRODIGY project: Daniel Borrajo, Oren Etzioni, Dan Kahn, Dan Kuokka, Michael Miller, Alicia Perez, William Reilly, Santiago Rementeria, Nobuyoshi Wada, and Xuemei Wang.

References

[Agre and Chapman, 1987] Philip E. Agre and David Chapman. Pengi: An implementation of a theory of activity. In *Proceedings of the National Conference on Artificial Intelligence.* Seattle, WA, 1987.

[Brooks, 1986] R. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1), 1986.

[Carbonell and Veloso, 1988] J. G. Carbonell and M. M. Veloso. Integrating derivational analogy into a general problem solving architecture. In *Proceedings of the First Workshop on Case-Based Reasoning*, pages 104–124, Los Altos, CA, May 1988. Morgan Kaufmann.

[Carbonell *et al.*, 1990] Jaime G. Carbonell, Craig A. Knoblock, and Steven Minton. Prodigy: An integrated architecture for planning and learning. In Kurt VanLehn, editor, *Architectures for Intelligence*. Erlbaum, Hillsdale, NJ, 1990.

[Etzioni, 1990] Oren Etzioni. Why PRODIGY/EBL works. In *Proceedings of Eighth National Conference on Artificial Intelligence*, Boston, MA, 1990.

[Joseph, 1989] Robert L. Joseph. Graphical knowledge acquisition. In *Proceedings of the 4th Knowledge Acquisition For Knowledge-Based Systems Workshop*, Banff, Canada, 1989.

[Knoblock, 1990] Craig A. Knoblock. Learning effective abstraction hierarchies. In *Proceedings of Eighth National Conference on Artificial Intelligence*, Boston, MA, 1990.

[Langley *et al.*, 1989] Pat Langley, Kevin Thompson, John A. Allen, Wayne F. Iba, and John Gennari. An integrated cognitive architecture for autonomous agents. Technical report, Department of Information and Computer Science, University of California, Irvine, 1989.

[Minton *et al.*, 1989] Steven Minton, Craig A. Knoblock, Daniel R. Kuokka, Yolanda Gil, Robert L. Joseph, and Jaime G. Carbonell. PRODIGY 2.0: The manual and tutorial. Technical Report CMU-CS-89-146, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, May 1989.

[Minton, 1988] Steven Minton. *Learning Effective Search Control Knowledge: An Explanation-Based Approach.* PhD thesis, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, 1988.

[Mitchell *et al.*, 1990] Tom M. Mitchell, John Allen, Prasad Chalasani, John Cheng, Oren Etzioni, Marc Ringuette, and Jeffrey C. Schlimmer. Theo: A framework for self-improving systems. In Kurt VanLehn, editor, *Architectures for Intelligence*. Erlbaum, Hillsdale, NJ, 1990.

[Rosenbloom *et al.*, 1990] Paul S. Rosenbloom, Allen Newell, and John E. Laird. Towards the knowledge level in SOAR: The role of the architecture in the use of knowledge. In Kurt VanLehn, editor, *Architectures for Intelligence*. Erlbaum, Hillsdale, NJ, 1990.

[Veloso and Carbonell, 1989] M. M. Veloso and J. G. Carbonell. Learning analogies by analogy the closed loop of memory organization and problem solving. In *Proceedings of the Second Workshop on Case-Based Reasoning*, pages 153–159, Los Altos, CA, May 1989. Morgan Kaufmann.

[Veloso, 1989] Manuela M. Veloso. Nonlinear problem solving using intelligent casual-commitment. Technical Report CMU-CS-89-210, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 1989.

Execution-time Response: Applying plans in a dynamic world

Kristian J. Hammond
Artificial Intelligence Laboratory
Department of Computer Science
The University of Chicago

Phil Agre
Artificial Intelligence Laboratory
Department of Computer Science
The University of Chicago

Richard Alterman
Department of Computer Science
Brandeis University

Reid Simmons
School of Computer Science
Carnegie Mellon University

R. James Firby
NASA: Jet Propulsion Labs

Abstract

This panel is aimed at the issue of how to use and modify plans during the course of execution. The relationship between a plan and the actions that an agent takes has generated a great deal of interest in the past few years. This is, in part, a result of the realization that planning in the abstract is an intractable problem and that much of the complexity of behavior is best understood in terms of the complexity of the environment in which that behavior occurs.

This panel presents five distinct personalities and approaches to this problem:

- Agre looks at replacing "planning" with situated activity. In particular, he has been considering the problems involved with the reference assumptions of classical planning.
- Firby's hierarchical planner has primitive actions that are instantiated at execution-time. The execution of these primitives generates information that can be used to guide selection of later operators.
- In Alterman's model of run-time adaptation, the executive responds to failures by using external cues to move between alternative steps or approaches stored in an existing network of semantic/episodic information.
- Simmons has been exploring techniques to create robust, reactive systems that can handle multiple tasks in spite of the robot's limited sensors and processors. His approach takes full advantage of the resources that the robot does have. This includes using hierarchical coarse-to-fine control strategies, using concurrency whenever feasible, and explicitly focusing attention on the robot's tasks and monitored conditions.
- Hammond suggests a theory of *agency* which casts planning as embedded within a memory-based understanding system connected to the environment. Within this approach, the environment, plan selections, decisions, conflicts and actions are viewed through the single eye of situation recognition and response.

The Role of Plans in Activity

Phil Agre
The University of Chicago
Artificial Intelligence Laboratory

The notion of a plan has long been central to computational research on action. The terminology and characteristic hypotheses of 'planning' research received their most influential early formulation in Miller, Galanter, and Pribram's book *Plans and the Structure of Behavior* (1960), henceforth MG&P. MG&P's central thesis was that the observable structure of an organism's behavior results from its executing a Plan which has that same structure. MG&P demonstrated that a wide variety of phenomena could be assimilated to this model. The first system which operated by constructing and executing plans was Strips, built by Fikes and Nilsson (1971) using the problem-space methodology developed by Newell and Simon (1963). A great deal of research has been conducted within this framework (Chapman, 1987; Georgeff, 1987), recently leading to industrial applications (Wilkins, 1988; 1989).

The argument of MG&P contains a profound ambiguity. It can be viewed as running together two different accounts of plans and execution. On the first account, plans are primarily retrieved from a library, or constructed from scratch, and are executed whole. This account offers an explanation for the structure of behavior, but it portrays the agent as almost entirely inflexible. On the second account, an agent assembles its plans incrementally, so that sequences of actions need not be mapped out ahead of time. This account offers an explanation of how an agent might be capable of dynamically adapting its behavior to circumstances as they arise, but it does not explain why an organism's behavior has its observed structure. MG&P's two accounts of planning might be compatible in some complex combination, but they cannot simultaneously explain both the observable structure and the flexible adaptation of behavior.

This ambiguity within MG&P's argument has been fateful for subsequent research, particularly in the last five years as various groups have worked to build agents which are capable of conducting sensibly organized goal-directed action in environments characterized by unpredictability, uncertainty, and change. Much of this work has been conducted within the vocabulary of plans and their execution, trying to find an acceptable combination of the two approaches that MG&P introduced (Georgeff and Lansky, 1987; Firby, 1987). Much other work, though, has dispensed with the notion of a plan altogether, treating continual interaction with the environment as a central phenomenon (Agre in preparation; Brooks, 1986). For this work, the observable structure of an organism's behavior is an emergent property of these interactions and not the causal product of the execution of a plan. Such is the approach that I developed in the notion of running arguments (Agre, 1985) and that David Chapman and I took in our work on the Pengi system (Agre and Chapman, 1987).

People regularly make and use plans, of course, in the ordinary vernacular sense of the word; the introduction of an alternative to MG&P's account of action reopens the question of what plans actually are. They are not like computer programs, since their use regularly involves a considerable amount of interpretive effort as well as rearrangement, interpolation, and substitution of the actions the plan represents. Plans are moreover not a unified phenomenon; they occur in a wide variety of activities and social contexts, from cooking (Scher, 1984) to office procedures (Suchman, 1983) to navigation (Gladwin, 1970). In each case, though, it is best to view plans as resources for the participants in an activity, and not (as with MG&P) as fully specifying or causally engendering the activity (Suchman, 1987).

In recent work, Chapman and I have begun exploring a view of plans as communication in natural language (Agre and Chapman in press). People who use directions or instructions to find subway stations or play video games interpret them within the cultural background that they share with other participants in the activity. Moreover, they interpret them indexically, in terms of their ongoing situations as they understand them at the moment they turn to the plan for advice. Computational research can be expected to offer insight on these processes by investigating the architectural consequences of various ways of using plans and learning from their use, but such research must be informed by sociologically sophisticated views of the circulation and deployment of representational materials in human societies.

Adaptive Planning[1]

Richard Alterman
Computer Science Department
Brandeis University

There were two key features of early models of planning in artificial intelligence. The first was that the planner did not have a memory of previous planning episodes. This meant that the planner was always planning from scratch, constructing a plan out of a small set of operators. Second, these early models of planning almost entirely separated the planning and acting phases. A robot given some task would construct a plan from scratch to achieve that task. Then it would turn the plan over to an execution monitor that would supervise the robot as it went through the steps of the plan. This model of planning and acting proved undesirable because, in general, it failed to provide for the contingencies that might arise. This critique of early models arose from work on case-based planning (Hammond, 1990 and Kolodner and Simpson, 1989), reactive planning (Firby, 1987) and situated activity (Agre and Chapman, 1987 and Suchman, 1987).

Adaptive planning (Alterman, 1988) was an early effort to deal with the problems of traditional models of planning. An adaptive planner is a common sense planner. It has a memory of previous plans (routines) and retrieves a plan from that memory that seems to match the situation-at-hand. It then adapts that plan (improvises) during the period of engagement. For example, the first time I ride the NYC Subway, I do not plan from scratch; rather, I use my knowledge of riding BART (Bay Area Rapid Transit) as a basis for constructing an interpretation of the actions I should take.

I am currently looking at reasoning about the usage of mechanical devices and the role of instructions. A planner may adapt a known routine to the situation-at-hand, but, if difficulties arise, it has access to instructions. The specific difficulty provides a concrete context for those instructions. Much of this work is informed by my work on semantic memory (Alterman, 1989) and by the lexical semantic theory of Pustejovsky (Pustejovsky, forthcoming) as it impacts spatial and deictic terms.

With Roland Zito-Wolf, I am also looking at extending the adaptive planning model to handle plan learning. We assume that planners have *habitats*—places where they normally plan and act (e.g. home, the office, hotels). Thus plan learning involves a mixture of teasing out descriptions of the planner's habitats while extending plans to cope with new contingencies. Over a history of engagements the planner gradually settles, and re-settles, into customized routines for its habitats. An important advantage of the adaptive planning framework for learning is that learning is fail-safe, since incomplete or incorrect learning is backed-up by normal adaptive functioning.

[1]This work was supported in part by the Defense Advanced Research Projects Agency, administered by the U.S. Airforce Office of Scientific Research under contract #F49620-88-C-0058.

Task Directed Adaptive Execution

Robert James Firby
Jet Propulsion Laboratory

A robot plan is usually viewed as a list of primitive robot actions which are assembled in advance and then executed one after another. However, in real domains, a plan must have more structure if it is to cope with the myriad unpredictable details that it will encounter during execution. Adding such structure to a plan involves more than augmenting the primitive plan representation; it requires a complete model of plan interaction with the world. A planner cannot know in advance all of the sensing and control actions that will be required to achieve its goals because it cannot maintain a complete, detailed model of the situations that it will encounter. Most sensing and control decisions must be suspended until execution time. Therefore, the notion of a plan no longer makes sense without a theory of how it will be executed.

The RAP adaptive execution system is a theory of plan representation and execution. The system assumes an incomplete world model and relies on program-like reactive action packages (RAPs) to carry out sketchy plans premised on that model. A plan consists of a list of tasks rather than primitive actions. Each task contains three major components: a satisfaction test, a window of activity, and a set of execution methods that are appropriate in different circumstances. Plan execution proceeds by selecting an unsatisfied task and choosing a method to achieve it based on the current known world state. A task may be executed as many times as necessary to keep it satisfied while it is active. Since decisions on action selection and sensor deployment are made while the task is "situated" in the real world, execution monitoring is an intrinsic part of the execution algorithm, and the need for separate replanning on failure disappears.

The RAP system appears to offer an effective way to cope with the limitations imposed by real sensors, real actuators, and the incomplete understanding of complex domains.

Robust Robots with Limited Resources

Reid G. Simmons
School of Computer Science
Carnegie Mellon University

A prevalent approach to building mobile robot systems is to have the system continually monitor all (relevant) aspects of the environment, and use what are essentially stimulus-response rules to decide what to do next. Getting the robot to perform a new task typically involves adding more sensors and/or processors. While this approach produces very reactive systems, it does not scale well as the tasks become more complex and numerous.

We are exploring techniques to create robust, reactive systems that can handle multiple tasks in spite of the robot's limited sensors and processors. To succeed, our approach tries to take full advantage of the resources that the robot does have. This includes using hierarchical coarse-to-fine control strategies, using concurrency whenever feasible, and explicitly focusing attention on the robot's tasks and monitored conditions.

We have developed the Task Control Architecture (TCA) to support the creation of such systems. TCA is a distributed system with central control that can construct and manipulate hierarchical plans, allocate and manage user-defined resources, monitor selected conditions, and handle

exceptions. Robot-specific processes (such as controllers, planners, and vision processes) communicate with one another through TCA and use the TCA mechanisms to schedule and synchronize their activities.

TCA is currently in use on two testbeds — a prototype of the six-legged CMU Planetary Rover, and the Hero, an indoor mobile manipulator, whose tasks include collecting cups from our lab's floor, retrieving printer output, delivering objects, and recharging itself when necessary. In implementing the Hero and Rover systems, several simple, but effective, organizing principles have emerged for taking full advantage of the robot's available resources.

Hierarchy: Hierarchy is effective for planning, monitoring, and handling exceptions. Our system plans only to the level of detail warranted by its current knowledge of the environment, deferring the remaining details to be filled in at execution time. The Hero system uses coarse-to-fine sensing strategies. For example, the system uses its 2D vision system to detect approximately cup-shaped regions, which triggers tasks to approach and map the objects with a wrist-mounted sonar to determine if they in fact match the robot's model of a cup.

Concurrency: The distributed nature of TCA is used to exploit opportunities for concurrency. These include interleaving of planning and execution, and asynchronous pre-processing of visual data. However, our systems do not have sufficient sensors or computational resources to continually monitor all conditions. TCA handles this by enabling processes to specify the temporal intervals during which selected conditions should be monitored, and the frequency at which they should be polled. For optimal performance, these frequencies should be based on the likelihood of the monitored condition occurring, the urgency for response, and the time needed to react.

Focus of Attention: Since it is unreasonable to expect the system to monitor all possible conditions or to plan for all tasks at once, the robot must explicitly maintain a focus of attention. The resource mechanism of TCA is used to maintain the focus of attention on the currently active tasks. Multiple tasks can be handled concurrently as long as they use separate resources; when contention for resources occurs, the associated tasks must be prioritized. While TCA currently uses only a simple, pre-programmed prioritization scheme, we are starting to explore how the robot could make its own prioritization decisions by reasoning about models of its capabilities, limitations, and the relative utility of the tasks. For example, if the Hero's battery monitor warns of a low charge, the robot should decide whether it can afford to complete its current task, or whether it should immediately proceed to the charger.

Memory and Agency[2]

Kristian J. Hammond
The University of Chicago
Artificial Intelligence Laboratory

Increasingly, the study of planning is being recast as the broader study of planning, action, and understanding. The particular approach that we are taking to this study casts planning as embedded within an understanding system connected to the environment. This approach allows us to view plan selection, conflict resolution, and action through the single eye of situation assessment. Together with the use of episodic memory as the vehicle for understanding, this view leads naturally

[2]This work was supported in part by the Defense Advanced Research Projects Agency, monitored by the Air Force Office of Scientific Research under contract F49620-88-C-0058, and the Office of Naval Research under contract N0014-85-K-010.

to an ability to learn from both planning and execution. In this paper, we draw an outline of *agency*, our model of the relationship between planning and action.

Our model of the relationship between planning and action is a complete theory of *agency*. Our theory of agency rises out of three pieces of work: Schank's structural model of memory organization (Schank, 1982), our own work in case-based planning and dependency-directed repair (Hammond, 1986), and the work of Martin and Riesbeck in Direct Memory Access Parsing (Martin 1989). Our model has been articulated in two programs, TRUCKER and RUNNER (Hammond, Marks, and Converse, 1988; Hammond, 1989).

Our original objective was to capture the ability of an agent to suspend goals, yet still recognize execution-time opportunities to satisfy them. We used a single set of memory structures both to store suspended goals and to understand the agent's circumstances in the world. In response to a blocked goal, the agent's first step was to do a planning-time analysis of the conditions that would favor the satisfaction of the goal. The agent then *suspended* the goal in memory, indexed by a description of those conditions.

During execution, the agent performed an ongoing "parse" of the world in order to recognize conditions for action execution. Following DMAP (Martin, 1989), this parse took the form of marker passing through episodic memory. Because suspended goals were indexed in the same memory used for understanding the world, the goals were activated when the conditions favoring their execution were recognized. Once active, goals would be reevaluated in terms of the new conditions. If they fit into the current flow of execution, they would be pursued. Otherwise, they would be suspended again.

We called the initial model *opportunistic memory* because the agent's recognition of opportunities depended on the nature of its episodic memory structures. Having turned to the broader issues of integrating planning and action, we now refer to our work as the study of *agency*.

Our theory of agency accounts for the spawning of goals, the selection of plans, and the execution of actions. Like DMAP, our theory relies on a memory organization defined by part/whole and abstraction relationships. Activations from features in the environment are passed up through abstraction links, and predictions are passed down through partially active concepts.

To accommodate action, we have supplemented DMAP with the notion of PERMISSIONS and POLICIES. PERMISSIONS are handed down the parts of plans to the operators they include. The only actions that take place are those that are PERMITTED by the activation of the operators that are associated with them. Following (McDermott, 1978), POLICIES are statements of ongoing goals of the agent. They may take the form of maintenance goals, such as "Glasses should be in the cupboard" or "Always have money on hand." The only way in which goals can be generated is out of the interaction between POLICIES and environmental features.

Most of the processing takes the form of recognizing circumstances in the external world in the context of the policies, goals and plans of the agent. Goals, plans, and actions interact with each other and with the environment as follows:

- Features in the environment interact with POLICIES to spawn goals.
- Goals and environmental features combine to activate plans already in memory.
- Operators are permitted by plans and are associated with the descriptions of the world states appropriate to their performance. Once a set of features has an operator associated with it,

that set of features (in conjunct rather than as individual elements) is now predicted and can be recognized.

- Operators are specialized into actions by features in the environment and by internal states of the system. As with Firby's RAPs (Firby, 1989), particular states of the world determine particular methods for each general operator.
- Conflicts between actions are recognized and mediated by the same mechanism that parses the world.
- Suspended goals are associated with the descriptions of the states of the world that are amenable to their satisfaction.

This theory of agency not only bridges the gap between planning and execution, but approaches a start-to-finish model of behavior in the world. Our goal is a content theory of agency. We see this architecture as simply the vehicle for that content.

References

Philip E. Agre, Routines, AI Memo 828, MIT Artificial Intelligence Laboratory, 1985.

Philip E. Agre and David Chapman, Pengi: An implementation of a theory of activity, *Proceedings of the Sixth National Conference on Artificial Intelligence*, Seattle, 1987, pages 196-201.

Philip E. Agre and David Chapman, What are plans for?, in Pattie Maes, ed., *New Architectures for Autonomous Agents: Task-level Decomposition and Emergent Functionality*, MIT Press, Cambridge MA, in press.

Philip E. Agre, *The Dynamic Structure of Everyday Life*, Cambridge University Press, Cambridge, in preparation.

Richard Alterman. Event Concept Coherence. *Semantic Structures*, Editor David Waltz, LEA, 1989.

L. Birnbaum and G. Collins. Opportunistic planning and Freudian slips. In *Proceedings of the Sixth Annual Conference of the Cognitive Science Society*, Boulder, CO, 1984.

Rodney A. Brooks, A robust layered control system for a mobile robot, *IEEE Journal of Robotics and Automation* **2**(1), 1986, pages 14-23.

David Chapman, Planning for conjunctive goals, *Artificial Intelligence*, **32**(3), 1987, pages 333-377.

Richard Fikes and Nils Nilsson, Strips: A new approach to the application of theorem proving to problem solving, *Artificial Intelligence* **2**(3), 1971, pages 189-208.

R. James Firby, An investigation into reactive planning in complex domains, *Proceedings of the Sixth National Conference on Artificial Intelligence*, Seattle, 1987, pages 202-206.

R. James Firby. Adaptive execution in complex dynamic worlds. Research Report 672, Yale University Computer Science Department, 1989.

Michael P. Georgeff, Planning, in Joseph F. Traub, Barbara J. Grosz, Butler W. Lampson, and Nils J. Nilsson, eds, *Annual Review of Computer Science* **2**, Palo Alto, CA, 1987, pages 359-400.

Michael P. Georgeff and Amy L. Lansky, Reactive reasoning and planning, *Proceedings of the Sixth National Conference on Artificial Intelligence*, Seattle, 1987, pages 677-682.

Thomas Gladwin, *East is a Big Bird*, Harvard University Press, Cambridge MA, 1970.

Kristian Hammond. *Case-Based Planning: Viewing Planning as a Memory Task*, volume 1 of *Perspectives in Artificial Intelligence*. Academic Press, 1989.

Kristian Hammond. Opportunistic memory. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*. IJCAI, 1989.

Kristian Hammond. Explaining and repairing plans that fail. *Artificial Intelligence Journal*, In Press.

Kristian Hammond, Tim Converse, and Mitchell Marks. Learning from opportunities: Storing and reusing execution-time optimizations. In *Proceedings of the Seventh Annual Conference on Artificial Intelligence*, pages 536–40. AAAI, 1988.

Janet Kolodner and Robert Simpson. The MEDIATOR. *Cognitive Science Journal,* 13:507-549, 1989.

Charles E. Martin. *Direct Memory Access Parsing.* PhD thesis, Yale University Department of Computer Science, 1989.

D. McDermott. Planning and acting. *Cognitive Science*, 2, 1978.

George A. Miller, Eugene Galanter, and Karl H. Pribram, *Plans and the Structure of Behavior*, Henry Holt and Company, New York, 1960.

Allen Newell and Herbert A. Simon, GPS: A program that simulates human thought, in Edward A. Feigenbaum and Julian Feldman, eds, *Computers and Thought*, McGraw-Hill, New York, 1963, pages 279-296.

James Pustejovsky. *The Generative Lexicon.* MIT PRESS, forthcoming.

Roger C. Schank. *Dynamic Memory: A Theory of Reminding and Learning in Computers and People.* Cambridge University Press, 1982.

Bob Scher, *The Fear of Cooking*, Houghton-Mifflin, Boston, 1984.

Lucy Suchman, Office procedures as practical action: Models of work and system design, *ACM Transactions on Office Information Systems* 1(4), 1983, pages 320-328.

Lucy Suchman, *Plans and Situated Action*, Cambridge University Press, Cambridge, 1987.

David E. Wilkins, *Practical Planning: Extending the Classical AI Planning Paradigm*, Morgan Kaufmann Publishers, Los Altos CA, 1988.

David E. Wilkins, Can AI planners solve practical problems?, Technical Report 468, SRI International Artificial Intelligence Center, 1989.

NEONATE COGNITION SYMPOSIUM

Chair: Richard Held, Department of Brain & Cognitive Sciences, MIT

Advances in experimental technique and technology reveal more and more hitherto unexpected cognitive capabilities in infants. Moreover, these achievements appear at earlier and earlier ages. We discern a trend in infant research as investigators increasingly attribute the appearance of cognitive capacity to the development of sensory and motor systems as opposed to that of higher centers in the brain. Pushed to an extreme this trend leads to the view that the neonatal mind-brain lacks only the sensory input and motor output required to potentiate its capability. In place of the doctrine of the neonate tabula rasa, organized by sense and movement, we would have the neonate tabula cognitiva awaiting the perfection of its sensory and motor apparatus in order to engage the world. Echoing this theme, Jane Gwiazda will discuss the development of vision in infants. She reports that visual resolution, limited initially by peripheral retinal processes, improves slowly over several years. In contrast, the hyperacuities, with presumedly more significant central cortical components, rise much more abruptly. Renee Baillargeon presents evidence that very young infants understand that objects possess at least some of the physical properties taken for granted by adults. Their perceptual worlds are structured in accord with this understanding. Quantitative physical reasoning is a bit slower to develop but it too is present by the end of the first half-year. Adele Diamond has discovered that infants understand concepts long before they have the capacity to plan and inhibit the motor activity required to perform responses during testing. Sequential actions requiring motor planning and reflex inhibition do not appear until half a year of age. Together these reports argue for the change in viewpoint discussed above.

Discussant: Jacques Mehler

DEVELOPMENT OF VISION

Jane Gwiazda, Department of Brain & Cognitive Sciences, MIT

Study of the development of vision provides an opportunity to relate changes in visual functions to changes in the nervous system. Over the past fifteen years we have studied basic measures of visual function, including grating acuity, stereoacuity, and vernier acuity, in human infants. An initial attempt has been made to relate the developmental changes in vision to changes in the underlying neuronal mechanisms.

All of our data were obtained using a two-alternative forced choice preferential looking procedure. Grating acuity develops more slowly than vernier and stereoacuity. It is very poor at birth and does not reach adult levels until 3 or 4 years of age. Changes occurring at the retinal level during the early months can account for the increases in grating acuity.

Vernier acuity is termed a hyperacuity because the smallest offset of a line that can be detected by an adult is almost an order of magnitude more sensitive than the minimal separation of foveal cones. It is thought by many to be cortically processed. In infants, vernier acuity is actually poorer than grating acuity in the first two months, but rises rapidly and exceeds grating acuity by the fourth month.

Stereopsis is not present in very young infants, but has an abrupt onset at 3 to 4 months. We have speculated that the onset is dependent upon the segregation of ocular dominance columns in layer 4 of the cortex. Stereoacuity rises very rapidly, approaching adult values within a few weeks from onset of coarse stereopsis. This implies fine tuning of the disparity-sensitive neuronal mechanism.

Females show superiority over males in months 4 through 7 on the hyperacuities, but not grating acuity, which suggests that sex differences are found at the cortical level.

YOUNG INFANTS' PHYSICAL WORLD

Renee Baillargeon, Department of Psychology, University of Illinois

Infants' understanding of the physical world has traditionally been characterized as severely limited. Infants are said to hold no beliefs or to hold incorrect beliefs about objects, and to possess at best primitive physical reasoning abilities. However, recent evidence collected by my collaborators and myself suggests that this characterization is inaccurate. My talk is divided into three sections. In the first section, I report experiments that indicate that young infants share many of adults' beliefs about the physical world. To illustrate, I argue that, by 3.5 months of age, infants understand that (a) objects continue to exist when out of sight; (b) objects cannot move through the space occupied by other objects; and (c) objects fall when their supports are removed. In the second section, I summarize experiments that suggest that young infants possess an impressive array of physical reasoning abilities. I distinguish between qualitative reasoning (e.g., determining that a rotating screen should stop when encountering an obstacle in its path) and quantitative reasoning (e.g., judging at what point a rotating screen should stop based on the location and height of the obstacle in its path), and argue that infants engage in the former type of reasoning several months before they do the latter. In the third section, I present experiments that reveal how infants' physical knowledge affects their perceptions and categorizations of objects. I conclude with a review of the many similarities and few differences brought to light by the present research between infants' and adults' physical worlds.

THE PLANNING, EXECUTION, AND INHIBITION OF MOVEMENT DURING INFANCY

Adele Diamond, Department of Psychology, University of Pennsylvania

Infants appear to understand the concept of contiguity, understand and remember that an object they can no longer see is still there, and remember which of two objects they have already seen, long before they can show these abilities on traditional measures. Why?

I would like to suggest that traditional measures require planning of motor actions and inhibition of motor actions, both of which require maturation in the supplementary motor area (SMA). Maturation in SMA may account for many of the advances we see in infants' behavior between 5-8 months of age. Note, the suggestion is that the cognitive abilities mentioned in paragraph 1 are present early, but they cannot be demonstrated in reaching behavior until much later.

SMA is required for putting two or more actions together into a sequence, as for example, in uncovering one object to retrieve another or in executing a circuitous reach with two-vector components. It is also required for the inhibition of the reflexes of the hand: the grasp and avoidance reflex. Traditional measures of object permanence (uncovering a hidden object), recognition memory (the delayed non-matching to sample task), and spatial contiguity (retrieving an object directly behind a screen) have all required skills dependent on SMA. Tasks using looking as the dependent measure have not required motor planning or inhibition and have shown these memory and conceptual abilities to be present much earlier; similar results have recently been obtained with reaching tasks that do not require SMA-dependent abilities.

Perception, Computation, and Categorization

Whitman Richards

M.I.T.

The thread through these three seemingly somewhat disjoint topics is the role features and models play in perceptual categorizations. Bobick begins by proposing that perceptual categorization is useful for predicting unobserved properties of objects from their observable properties. Such properties, of course, are inferred from features derived from our sense data. Which of the infinity of possible features should we elect to construct to support these inferences? Bobick provides an answer to this question by showing how a particular model for structure in the world can drive both feature selection and categorization. Nakayama then explores one very special case of the class of models proposed by Bobick namely the proposal that objects occupy distinctly separate regions in space and time. The predictable, but unobserved property is the surface of one object that is hidden or occluded by another. Nakayama shows that the machinery supporting such inferences is located quite early in the visual pathway. Finally, Jepson focusses on how feature maps interface with models such as those required for making inferences about occlusion, thereby allowing one to assign the proper world structure to seemingly inadequate sense data. The result of this interplay between features and models is an ordering of possible categorizations of the data, which leads to a formal definition for the categorization selected as "the perception."

Natural Object Categorization and Perceptual Inference

Aaron Bobick

SRI International

We present a basis for generating natural categories of objects. To provide a criteria for categorization we propose that the purpose of a categorization is to support the inference of unobserved properties of objects from the observation of perceivable properties. Because no such set of categories can be constructed in an arbitrary world, we make a claim about the structure of the world -- the Principle of Natural Modes -- and argue that a natural categorization should to reflect this structure.

We first define an evaluation function that measures how well a set of categories supports the inference goals of the observer. Entropy measures for property uncertainty and category uncertainty are combined through a free parameter that reflects the goals of the observer. Natural categorizations are shown to be those that are stable with respect to this free parameter. These categories are related to Rosch's Basic Level Categories. We next develop a categorization paradigm that utilizes the categorization evaluation function in recovering natural categories. A statistical hypothesis generation algorithm is presented that is shown to be an effective categorization procedure. Examples drawn from several natural domains are presented, including data known to be a difficult test case for numerical categorization techniques. Once a categorization has been discovered, the evaluation function can be used to measure the utility of features in recovering natural categories. Finally, we consider the extension of the paradigm to a categorization system that describes objects using a representation language richer than simple property descriptions.

Visual Inference in the Perception of Occluded Surfaces

Ken Nakayama

Harvard University

If a surface is hidden behind another surface, we often infer its continued existence even though it is literally invisible.

At what level of cognitive representation should we attribute such knowledge of occlusive spatial relations? Is our knowledge of occlusion represented relatively late, perhaps as a form of general spatial reasoning, or is it specifically visual and represented much earlier?

From a series of perceptual demonstrations and psychophysical experiments, we argue that the processing of occlusion is specifically visual. Such experiments show the importance of occlusion related processing in the perception of form, motion, depth and color. Furthermore, we argue that in some cases, the processing of occlusion might begin as early as primary visual cortex (area V1).

What is a Perception?

Allan Jepson

University of Toronto

Perception is the subject of extensive study in AI, Neurobiology and Psychology. Yet the event itself that we call a perception has never been defined formally. Here, we offer a definition hinged around how our beliefs and knowledge of the world affect the interpretation of our sense data. Given a set of beliefs, a lattice of possible belief states can be created, where the nodes in the lattice are labelled by the beliefs which remain valid or invalid. A perception can then be defined as a local maximum in a partial ordering within this lattice. Two examples illustrate how our lattice theory can represent plausible perceptual states. In one example, we also introduce the notion of a "key feature" - one capable of supporting especially strong inferences. Our "lattice theory" suggests a more meaningful, higher level state of understanding for machine systems, and also provides insight into human perception.

Principle-Based Parsing: A Symposium

Robert C. Berwick
MIT Artificial Intelligence Laboratory

Principle-based parsing, sometimes called principles-and-parameters parsing, is a relatively new approach to language analysis that replaces the traditionally large set of rules used to analyze sentences on a language-by-language basis with a much smaller, fixed set of parameterized, universal principles. The principles interact deductively to replace many rules. In such a system there is no "rule" of passive, encoded perhaps in context-free rule, if-then format, or a transition network; instead, there are broader axioms from which one can "deduce" the possibility of a passive-type sentence. Thus the shift is much like that has occurred in expert systems from a shallow if-then form to an explanatory system that reasons from first principles. The underlying linguistic theory is most often associated with the so-called "principles and parameters" framework or government-binding (GB) theory of linguistic analysis.

But this shift from rules to principles raises many serious questions for cognitive science. Is principle-based parsing possible? Can it really work universally, for many different languages, thus raising the possibility of machine translation once again? How can efficient parsing be done with general principles? What implementations are possible? How do alternative machine architectures fit in, including connectionism? In the five symposium papers that follow, and a general discussion we hope to address these and other problems in this rapidly developing area of natural language processing.

Principle-Based Parsing

Steven P. Abney
Bell Communications Research

I consider characterizations of principle-based parsing or government-binding theory parsing (GB parsing) that have been made in the literature. A common theme is the *transparency hypothesis*, according to which there is a close correspondence between the theoretical constructs of GB syntactic theory and either the data objects manipulated by the parser, or the procedures that make up the parser. There are at least three different motivations for the transparency hypothesis: (1) because it gives substance to the conviction that syntactic constructs represent real objects, and are not just convenient fictions, (2) because it makes program verification easier, (3) because it serves as a source of inspiration for new parsing techniques.

Motivation (1) is unjustified. There are many objects in the domain of parsing that are real, in the sense of providing the best explanation of surface phenomena, but which are neither data objects nor procedures. For example, in an LR parser generated from a grammar G, there is no data object or procedure that can be identified as a rule of G; the rules of G exist only in their combined consequences for the LR tables. Nonetheless, the best explanation of the behavior of the parser is in terms of the rules of G.

Motivation (2) is valid, but program verification is only one concern of many in software engineering. If (2) is our only motivation, then current work in principle-based parsing accords disproportionate weight to program verification.

Motivation (3) can hardly be faulted, though it makes the transparency hypothesis of value only for those people whom it inspires. I consider some GB-inspired techniques that have value independent of their relation to GB.

Principle-Based Machine Translation

Bonnie J. Dorr
MIT Artificial Intelligence Laboratory

This talk will describe UNITRAN, an implemented principle-based machine translation system. The task of cross-linguistic translation is difficult because there are several types of phenomena within any given language; moreover, the number of ways in which these phenomena can be exhibited is potentially enormous across different source and target languages. The primary characteristic of UNITRAN is that it operates uniformly across English, Spanish, and German, while still accounting for knowledge that is specific to each language.
Consider the following translation from English to Spanish:

John broke into the room ⇒ Juan forzó la entrada al cuarto

In this example, the source-language sentence diverges both syntactically and lexically from the target-language sentence. The syntactic divergence shows up in the translation of the single-object construction *into the room* into the double-object construction *la entrada al cuarto*. The lexical divergence shows up in the realization of the simple verb *break* as the composite form *forzar la entrada* (literally, *force entry*). The UNITRAN system solves these types of divergences by making use of two levels of processing: a syntactic level based on principles and parameters, and a lexical-semantic level based on the compositionality of lexical items.

UNITRAN differs from other translation systems in that it is able to maintain language-independent information while still processing many types of language-specific phenomena associated with different source and target languages. Within this framework, language-independent conceptual representations are used as the pivot between the source and target languages, and language-specific phenomena are accounted for by means of parameter settings and lexically-specified information.

The Computational Implementation of Principle-based Parsers

Sandiway Fong
MIT Artificial Intelligence Laboratory

Recently, there have been some interest in the implementation of grammatical theories based on the principles and parameters approach. In this framework, a fixed set of universal principles interact deductively to account for diverse linguistic phenomena.

In this talk, I will address the issue of how to organize such principles for efficient processing. That is, what is an appropriate 'machine architecture' for parsers based on the principles-and-parameters framework? I will discuss two possible approaches. First, I will discuss the design of parsers that build complete phrase structure representations and then apply principles in order to rule out those that are ill-formed. In this approach, I will discuss whether it is possible to exploit the relatively-free interaction of linguistic principles to engineer more efficient parsers. In particular, I will show that the hitherto largely ignored issue of principle ordering can have a large effect on parsing efficiency, and hence, should be of much concern to parser designers.

An alternative approach is to interleave the task of building phrase structure representations with principle application. I will address the question: Is principle interleaving a more effective parsing strategy than a non-interleaved approach? I will discuss both the conditions under which interleaving may be a more effective strategy, and circumstances where a non-interleaved approach may be preferable.

I will present the results of various experiments on different machine configurations in the context of an implemented system that was built to investigate these issues. This system allows the automatic construction of a whole family of parsers with different architectures that all use the same underlying set of principles.

Constraint-oriented Principle-based Parsing

Mark Johnson
Brown University

One of the difficulties in the construction of computational models for Chomsky's "Government and Binding" theory is that the appropriate representations of an utterance at different linguistic levels are determined by a system of tightly interacting principles. Simple-minded implementations which construct these levels of representation one by one cannot fully exploit the constraints on the first levels of representation they construct imposed by other as yet unconstructed levels: a disastrous short-coming since these reduce an infinite search space to a finite one. This problem can be avoided by determining in advance some or all of the constraints that the later, as yet unconstructed levels of representations will impose the other levels. It leads to a constraint-oriented rather than representation-oriented perspective: the construction of one representation requires only the constraints originating from the other representations, not the representations themselves. If *all* constraints originating from some level of representation can be determined without that level of representation, then that level need not be constructed by the computational model. Simple natural language parsers will be described that function in exactly this way, avoiding the explicit construction of several levels of representation, although provably producing analyses that satisfy all constraints on all linguistic levels (including those unconstructed).

Such computational models pose interesting problems for the "mental representation as data structure" analogy common in cognitive science, since arguably the computational model does exploit all of the levels of representations of an utterance (where else could the relevant constraints come from?), even though it does not construct data structures corresponding to each level of representation.

Relaxation Techniques for Principle-Based Parsing

Edward P. Stabler, Jr.
UCLA Dept of Linguistics

Consider parsers which are "principle-based" in the following sense: structural representations are obtained by deductive reasoning from an explicitly represented linguistic theory. Such parsers can still differ along many dimensions. For linguists and psychologists interested in the bearing of linguistic theory on accounts of human parsing, the differences between the following principle-based views are significant, where Con(G) is the set of consequences of grammar G:

> (G) Parsing from first principles: the linguists' grammar G is represented and used directly
>
> (EG) Parsing from equivalent principles: using a theory EG such that Con(EG)=Con(G)
>
> (WG) Parsing from weaker principles: using a theory WG such that Con(WG) is a proper subset of Con(G)
>
> (DG) Parsing from different principles: using a theory DG such that Con(D) is not a subset of Con(G).

Focusing on government-and-binding theory, it will be argued that (G) has serious complexity problems that can be avoided by a logically equivalent, "relaxed" approach (EG), which solves parsing problems after first solving appropriate approximations. A version of (EG) together with a plausible performance story would remove the motivation for going to (WG); and (DG) is the least appealing option, only of interest if the other options fail, and they haven't. I will conclude by considering whether the presented argument against (G) depends on assumptions that have been effectively challenged by connectionist approaches to constraint satisfaction.

Workshop on Recent Results in Formal Learning Theory

A Survey of Recent Results in Formal Learning Theory

Kevin T. Kelly

Formal learning theory provides a general framework for determining when it is possible for an agent to converge to the truth in each of a collection of possible circumstances. The theory was originally applied to questions about the learnability of classes of of languages, but recently the applications have been broadened to include automatic programming, theory discovery and other topics.

This lecture will present a general introduction the framework and some recent results of the theory, focussing on stuctural features of solvable and unsolvable discovery problems, and ways to apply the results to artificial intelligence and Bayesian learning.

Possibility and Impossibility Theorems for Cognitive Science

Clark Glymour

Learning theory provides a number of results about the possibility or the impossibility of solving fundamental discovery problems that arise in cognitive science. This lecture will describe applications to

*predicting the behavior of a "black box" containing an unknown arbitrary Turing machine--in general, you can't;
*predicting the behavior of a "black box" containing an arbitrary unknown finite automaton--in general, you can;
* deciding whether or not a system is Turing computable from its input/ouput behavior--in general, behavior cannot distinguish between computable and uncomputable systems.
* inferring cognitive architecture in cognitive neuropsychology from the capacities and incapacities of subjects with brain injuries.

SELF-ORGANIZING COGNITIVE AND NEURAL SYSTEMS

Stephen Grossberg, Organizer

This symposium explores recent neural network modelling results that clarify how humans and animals achieve autonomous real-time behavior that is remarkably adaptive in changing environments even if no teacher is available to explicitly characterize when changes occur or what rules they obey. The processes whereby behavior adapts autonomously in response to unpredictable environmental changes is often called *self-organization.* The results reported in the three symposium talks will explore cognitive and neural designs for self-organizing perception, cognition, and action systems.

How do perception, cognition, and action differ from one another? Each talk will identify different environmental problems and computational constraints that these systems need to satisfy. The results show that self-organization is a much more comprehensive concept than learning. Each talk will describe a specialized type of neural architecture that provides a natural realization of a different computational theory. Whereas learning plays a role in these architectures, the learning laws are specialized to interact in a functionally meaningful way with the specialized information processing properties of each architecture. The functional significance of such learning in behavior arises in the form of emergent, or interactive, properties of the architecture as a whole.

Despite the differences between these architectures, they also share fundamental processing ingredients. For example, on the level of cellular mechanism, all the systems utilize shunting cooperative-competitive interactions at one or more processing stages; on the level of modular design, gated dipole opponent processes play a key role, including their properties of transmitter habituation and antagonistic rebound; on the level of system design, principles of uncertainty, complementarity, symmetry, and resonance are notable. For example, the splitting of system designs into complementary specific and nonspecific subsystems is generally observed, with nonspecific Attentional Gain Control playing a role in attentive recognition learning that is in many ways homologous to the nonspecific Go mechanism that controls "acts of will" and overall speed of goal-oriented arm and speech articulator movements. Likewise, a specific learned cognitive expectation, or Critical Feature Pattern, plays a role in attentive recognition learning that may be compared to a specific learned motor expectation, or Target Position Code, in goal-oriented movements. Issues of general importance in cogntive science are also probed, such as differences between preattentive versus attentive processing, and conscious versus unconscious processing.

SELF-ORGANIZING NEURAL NETWORKS FOR VISUAL FORM AND MOTION PERCEPTION

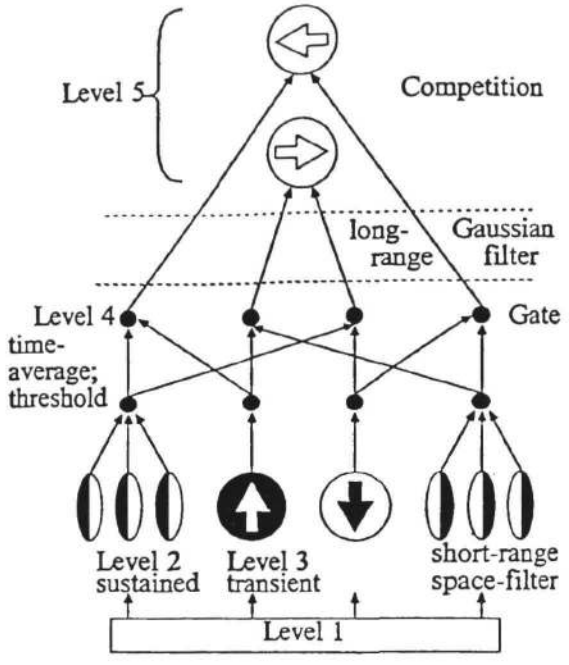
Stephen Grossberg, Ennio Mingolla, and Michael Rudd
Center for Adaptive Systems, Boston University,
111 Cummington Street, Boston, MA 02215

Why do parallel cortical systems exist for the perceptual processing of static visual forms and moving visual forms, and how do they work? These long-standing problems in perceptual and cognitive science have recently been clarified a as part of a theory of biological vision that is called FACADE Theory. FACADE Theory clarifies how the visual cortex generates a multiplexed representation of Form-And-Color-And-DEpth, or FACADE. This process is controlled by interactions of two systems, called the Boundary Contour System (BCS) and the Feature Contour System (FCS), whose properties are computationally complementary. The BCS generates an emergent 3-D boundary segmentation, whereas the FCS discounts the illuminant and fills-in surface properties. BCS segmentations are insensitive to direction-of-contrast, so that they can detect boundary structure along contrast reversals. They therefore do not generate perceived contrast differences within the BCS. FCS properties are sensitive to direction-of-contrast, and subserve visible percepts of surface brightness, color, and depth. The BCS includes striate and prestriate projections of cortical hypercolumns. The FCS includes striate and prestriate projections of cortical blobs (Grossberg, 1988; Grossberg and Mingolla, 1985a, 1985b).

The BCS is further divided into parallel systems for the processing of static segmentations and moving segmentations. The static BCS generates segmentations that are insensitive to direction-of-contrast and insensitive to direction-of-motion. Segmentations within the motion BCS are also insensitive to direction-of-contrast but sensitive to direction-of-motion. The motion BCS has been used to analyse a large body of data concerning short-range and long-range motion perception, notably data about apparent motion, such as beta, phi, gamma, delta, split, and Ternus motion; visual inertia; Korte's Laws; scale-dependence of D_{max}; form-color-stereo interactions; and competitive selection of motion pathways. Figure 1 schematizes the Motion Oriented Contrast (MOC) Filter for visual motion preprocessing (Grossberg and Rudd, 1989). The full model, which includes a cooperative-competitive feedback network, or CC Loop, suggests how coherent global motion effects occur, such as motion capture, induced motion, cooperative linking of oriented receptive fields, and cortical resonance. It also offers a solution of the global aperture problem (Grossberg and Mingolla, 1990).

A further analysis suggests why parallel cortical systems for analysis both of static form (V1 $\rightarrow$ V2) and of moving form (V1 $\rightarrow$ MT) exist (Grossberg, 1990), and how they generate different geometries of static and motion perception. These parallel systems compute all possible ways of symmetrically gating sustained cells with transient cells to generate output signals that are insensitive to direction-of-contrast and are organized into opponent pairs of on-cells and off-cells (see Figure 2). Opponent cell pairs in the static BCS define *orientations* that differ by 90° as opposites, whereas opponent cell pairs in the motion BCS define *directions* that differ by 180° as opposites. These opponent processes, called gated dipoles, are capable of habituation and antagonistic rebounds. Negative afterimages, such as circular afterimages of radial images and motion aftereffects such as the waterfall illusion, are clarified by their antagonistic rebound properties. These antagonistic rebounds act to rapidly reset previously active emergent segmentations, which could otherwise resonate for

Supported in part by the Air Force Office of Scientific Research (AFOSR F49620-87-C-0018), the Army Research Office (ARO DAAL-03-88-K0088), and DARPA (AFOSR 90-0083).



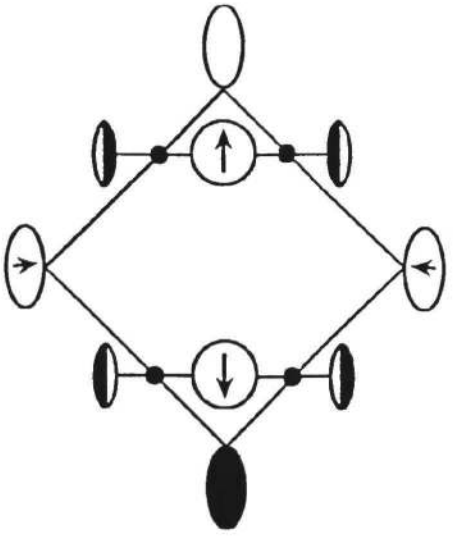


Figure 1 (left). The MOC Filter model for visual motion preprocessing.
Figure 2 (right). Four-fold symmetry of orientation cells and direction cells.

a long time after input offset or change, thereby causing massive smearing of percepts in response to rapidly changing images.

These results collectively argue against vision theories that espouse independent processing modules. Instead, specialized subsystems interact to overcome computational uncertainties and complementary deficiencies (Figure 1), and to realize symmetry principles (Figure 2) that are predicted to govern the development of visual cortex.

References

Grossberg, S. (Ed.) (1988). **Neural networks and natural intelligence**. Cambridge, MA: MIT Press.

Grossberg, S. (1990). Why do parallel cortical systems exist for the processing of static form and moving form? Submitted for publication.

Grossberg, S. and Mingolla, E. (1985a). *Psychological Review*, **92**, 173–211.

Grossberg, S. and Mingolla, E. (1985b). *Perception and Psychophysics*, **38**, 141–171.

Grossberg, S. and Mingolla. E. (1990). In M. Caudill (Ed.), **Proceedings of the international joint conference on neural networks, I**. 11–14. Hillsdale, NJ: Erlbaum.

Grossberg, S. and Rudd, M. (1989). *Neural Networks*, **2**, 421–450.

Self-Organizing Neural Network Modules for Control of Planned and Reactive Movements

Daniel Bullock & Stephen Grossberg
Center for Adaptive Systems
Boston University
111 Cummington Street
Boston, MA 02215 USA

A central goal of cognitive science is to understand the physical bases of behavioral operating characteristics, especially those associated with flexible intelligence. Among the most conspicuous and important of such operating characteristics are the perceptual constancies, because they raise fundamental issues of internal representation, and because of the difficulty of conceiving of human-like intelligence in the absence of such constancies. Less conspicuous but equally important for practical intelligence are movement control invariants, such as our ability to preserve the desired form – direction and endpoint – of a movement despite willed variations in movement speed, or our ability to hold a desired posture despite willed variations in the compliance of our limbs. An increasingly popular and very natural "language" for specifying quantitative theories of the physical bases of behavioral operating characteristics is the neural network formalism. Neural networks, when specified as systems of ordinary differential equations, allow qualitative and quantitative studies of components of intelligence as dynamical systems with rich emergent properties. This lecture presents two biologically constrained neural networks with the competence to ensure the movement control invariants mentioned above. The first is called the vector integration to endpoint, or VITE, model of variable speed trajectory formation. The second is called the FLETE (for factorization of length and tension) model of the spino-muscular system. Both models are intended to be simultaneously evaluated against all relevant physiological and motor psychophysical data, and a sample of such data will be compared with model structure and function in the course of the lecture.

In particular, properties of the VITE model are systematically compared with human and primate data on trajectories of planned point-to-point movements performed at variable speeds, and the model is shown to make correct predictions regarding all key kinematic signatures of such movements. For example, in addition to well-known empirical relations such as the speed accuracy trade-off discovered by Fitts (1954), the model correctly predicts the existence of a duration-dependent asymmetry in velocity profiles (e.g., Nagasaki, 1989). The existence of such an asymmetry in both planned arm and speech articulator movements suggests that the model may be applicable to a large range of planned movements, in which final execution of a stored movement goal is "gated" by an internal activating signal. The internal structure and function of the VITE model, including temporal dynamics of model cell types, are compared with neuroanatomical and neurophysiological data from primate studies, particularly studies of motor cortex and the basal ganglia.

The structure of the FLETE model is systematically compared with data on several further components of the mammalian neuromuscular system, especially spinal and cerebellar components. The model explains how we are able to continuously vary the compliance of our joints without inadvertently changing joint angle. This is important because of the key role of compliance control in modern approaches to human (Humphrey and Reed, 1983)

Supported in part by the National Science Foundation (NSF IRI-87-16960).
Supported in part by the National Science Foundation (NSF IRI-87-16960) and the Air Force Office of Scientific Research (AFOSR F49620-86-C-0037 and AFOSR F49620-87-C-0018).

as well as robotic (Hogan, 1980) movement. In humans and other animals, *independent* control of joint compliance and joint position is remarkable because muscle length remains invariant during a rescaling of muscle force, despite the natural tendency for muscle force to covary with muscle length. Moreover, mathematical analysis shows that if muscle tissue is subject to yielding at high force levels, then achieving a wide force range at each muscle length requires that motor units behave according to what is known empirically as the size principle of motor unit recruitment. In a system that obeys the size principle, attempting to vary joint compliance by adding the same co-contractive signal to both opponent muscle control channels disrupts invariant muscle length control, but the disruption may be preemptively compensated by an efference copy feedback pathway. A pathway with appropriate properties is provided in the spinal cord by Renshaw cells and Ia interneurons, and our simulations verify the compensatory properties of such a pathway. Further simulations show how a gradual learning process based on stretch feedback can allow the system to achieve feedforward compensation for changes, associated with joint rotations, in muscle mechanical advantage. In summary, the FLETE model explicates the neural bases of compensation for several intrinsic sources of variability in the realization of central motor commands, and thereby provides a rationale for many incompletely understood properties of the neuromuscular system.

Bullock, D. and Grossberg, S. (1988). Neural dynamics of planned arm movements: Emergent invariants and speed-accuracy properties during trajectory formation. *Psychological Review*, **95**, 49–90.

Bullock, D. and Grossberg, S. (1989). VITE and FLETE: Neural modules for trajectory formation and tension control. In W. Hershberger (Ed.), **Volitional Action**. Amsterdam: North-Holland, 253–297.

Bullock, D. and Grossberg, S. (1990). Adaptive neural networks for control of movement trajectories invariant under speed and force rescaling. *Human Movement Science*, **9**.

Fitts, P.M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, **47**, 381–391.

Grossberg, S. and Kuperstein, M. (1989). **Neural Dynamics of Adaptive Sensory-Motor Control: Expanded Edition**. Elmsford, NY: Pergamon Press.

Hogan, N. (1980). Mechanical impedance control in assistive devices and manipulators. **Proceedings of the Joint Automatic Control Conference**, San Francisco.

Humphrey, D.R. and Reed, D.J. (1983). Separate cortical systems for control of joint movement and joint stiffness: Reciprocal activation and coactivation of antagonist muscles. In J.E. Desmedt (Ed.), **Motor control mechanisms in health and disease**. New York: Raven Press, 347–372.

Nagasaki, H. (1989). Asymmetric velocity and acceleration profiles of human arm movements. *Experimental Brain Research*, **74**: 319–326.

ART: Self-Organizing Neural Networks for Learning and Memory of Cognitive Recognition Codes

Gail A. Carpenter and Stephen Grossberg
Center for Adaptive Systems, Boston University
111 Cummington Street, Boston, MA 02215

Adaptive resonance (ART) architectures are neural networks that self-organize stable pattern recognition codes in real-time in response to arbitrary sequences of analog or binary input patterns. In ART architectures, top-down learned expectation and matching mechanisms are critical in self-stabilizing the code learning process. A parallel search scheme updates itself adaptively as the learning process unfolds, and realizes a form of real-time hypothesis discovery, testing, learning, and recognition. A parameter called the attentional vigilance parameter determines how fine the categories will be. If vigilance increases (decreases) due to environmental feedback, then the system automatically searches for and learns finer (coarser) recognition categories. Learned representations are encoded in bottom-up and top-down adaptive filters whose long-term memory (LTM) traces vary slowly compared to the rapid short-term memory (STM) information processing.

Adaptive Resonance Theory emerged from an analysis of the instabilities inherent in feedforward adaptive coding structures (Grossberg, 1976a,b). More recent work has led to the development of three classes of ART neural network architectures, specified as systems of differential equations. The first class, ART 1, self-organizes recognition categories for arbitrary sequences of binary input patterns (Carpenter and Grossberg, 1987a). A second class, ART 2, does the same for either binary or analog inputs (Carpenter and Grossberg, 1987b). The talk will include ART 2 simulations that demonstrate how varying vigilance during learning can lead to the stable coexistence of coarse and fine category groupings that depend on the learning history, and show a non-uniform degree of discrimination across the set of inputs.

A third class, ART 3, solves computational problems of ART systems embedded in network hierarchies, where there can, in general, be either fast or slow learning and distributed or compressed code representations (Carpenter and Grossberg, 1990). ART 3 architectures incorporate a third memory, on an intermediate time scale, whose dynamics may be interpreted as chemical transmitter processes. The ART 3 medium-term memory (MTM) equations model the dynamics of production and release of a chemical transmitter substance; the inactivation of transmitter at postsynaptic binding sites; and the modulation of these processes via a nonspecific control signal. The net effect of these transmitter processes is to alter the ionic permeability at the postsynaptic membrane site, thus effecting excitation or inhibition of the postsynaptic cell. Specifically, the presynaptic signal, or action potential, S_i arrives at a synapse whose adaptive weight, or LTM trace, is denoted z_{ij}. The variable z_{ij} is identified with the maximum amount of available transmitter. When the transmitter at this synapse is fully accumulated, the amount of transmitter u_{ij} available for release is equal to z_{ij}. When a signal S_i arrives, transmitter is typically released. The variable v_{ij} denotes the amount of transmitter released into the extracellular space, a fraction of which is assumed to be bound at the postsynaptic cell surface and the remainder rendered ineffective in the extracellular space. Finally, x_j denotes the activity, or membrane potential, of the postsynaptic cell.

Acknowledgements: This research was supported in part by the Air Force Office of Scientific Research (AFOSR F49620-86-C-0037, AFOSR F49620-87-C-0018, and AFOSR 90-0128), the Army Research Office (ARO DAAL-03-88-K-0088), British Petroleum (89-A-1204), DARPA (AFOSR 90-0083), and the National Science Foundation (NSF DMS-90-00530 and IRI-87-16960).

Initially the transmitted signal pattern $\mathbf{S} \cdot \mathbf{u}_j$, as well as the postsynaptic activity x_j, are proportional to the weighted signal pattern $\mathbf{S} \cdot \mathbf{z}_j$ of the linear filter. The activity pattern of the target field is then contrast-enhanced, due to the internal competitive dynamics. The primary ART 3 MTM hypothesis assumes that the transmitter release rate is greatly amplified in proportion to the level of postsynaptic activity. A subsequent reset signal may thus selectively inactivate those pathways that cause an error. Following such a reset wave, the new signal $\mathbf{S} \cdot \mathbf{u}_j$ is no longer proportional to $\mathbf{S} \cdot \mathbf{z}_j$ but is, rather, biased against the previously active representation due to transmitter depletion at those sites. A series of reset events ensue, until an adequate match or a new category is found. Learning occurs on a time scale that is long relative to that of the search process.

The ART 3 MTM serves other functions as well as implementing the ART mismatch-reset-search cycle. In particular it allows the neural network to dispense with special processes to reset STM at onset or offset of an input pattern. The representation of input patterns as a sequence, $\mathbf{I}_1, \mathbf{I}_2, \mathbf{I}_3, \ldots$, corresponds to the assumption that each input is constant for a fixed time interval. In practice, an input vector $\mathbf{I}(t)$ may vary continuously through time. The input need never be constant over an interval, and there may be no temporal marker to signal offset or onset of "an input pattern" per se.

The ART 3 MTM transmitter depletion process can also serve to enhance features that were previously ignored. For example, suppose that the input ($\mathbf{I}$) signal pathways contain an ART 3 MTM process, and that a reset signal is generated, say, by an internal system error or by an external teaching input. Features represented in $\mathbf{I}$ that were not salient in the matched STM pattern $\mathbf{X}$ are enhanced, due to depletion in pathways leading to those features that, for whatever reason, generated an error signal. For instance, the previously ignored color of an object may be brought forth to enhance discrimination between category exemplars.

The mechanisms described thusfar are part of the recognition learning circuit of ART 3. Recognition learning is, however, only one of several processes whereby an intelligent system can learn a correct solution to a problem. We have called Recognition, Reinforcement, and Recall the "3 R's" of neural network learning (Carpenter and Grossberg, 1988). Various types of reaction to reinforcement feedback may be useful in applications. For example, a change in vigilance alters the overall sensitivity of the system to pattern differences; a shift in attention and the reset of active features can help to overcome prior coding biases that may be maladaptive in novel contexts.

In summary, the MTM provides the extra degree of freedom needed to embed ART systems in neural network hierarchies with fast or slow learning and compressed or distributed codes. The ART 3 MTM transmitter processes can also be used in a wide variety of fully or partially connected and adaptive or non-adaptive neural networks. The ART 3 search mechanism serves at least four distinct functions: to correct erroneous category choices; to learn from reinforcement feedback or disconfirmed expectations; to respond to changing input patterns; and, when as error signal occurs, to amplify features that were previously ignored. The talk will illustrate how ART modules embedded in neural network hierarchies carry out various target recognition functions.

References

Carpenter, G.A. and Grossberg, S. (1987a). A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, **37**, 54–115.

Carpenter, G.A. and Grossberg, S. (1987b). ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, **26**, 4919–4930.

Carpenter, G.A. and Grossberg, S. (1988). The ART of adaptive pattern recognition by a self-organizing neural network. *Computer*, **21**. 77–88.

Carpenter, G.A. and Grossberg, S. (1990). ART 3: Hierarchical search using chemical transmitters in self-organizing pattern recognition architectures. *Neural Networks*, in press.

Grossberg, S. (1976a). Adaptive pattern classification and universal recoding, I: Parallel development and coding of neural feature detectors. *Biological Cybernetics*, **23**, 121–134.

Grossberg, S. (1976b). Adaptive pattern classification and universal recoding, II: Feedback, expectation, olfaction, and illusions. *Biological Cybernetics*, **23**, 187–202.

Soar as a Unified Theory of Cognition: Spring 1990

Richard L. Lewis	School of Computer Science, Carnegie Mellon University
Scott B. Huffman	Department of Electrical Engineering and Computer Science, University of Michigan
Bonnie E. John	School of Computer Science, Carnegie Mellon University
John E. Laird	Department of Electrical Engineering and Computer Science, University of Michigan
Jill Fain Lehman	School of Computer Science, Carnegie Mellon University
Allen Newell	School of Computer Science, Carnegie Mellon University
Paul S. Rosenbloom	Information Sciences Institute, University of Southern California
Tony Simon	Department of Psychology, Carnegie Mellon University
Shirley G. Tessler	School of Computer Science, Carnegie Mellon University

Soar is a theory of cognition embodied in a computer system. In 1987 it was used as the central exemplar to make the case that cognitive science should attempt unified theories of cognition (UTC) [13][1]. Since then, much research has been done to move Soar toward being a real UTC, rather than just an exemplar. Figure 1 lists the relevant studies[2]. They have been done by a broad community of researchers in the pursuit of a multiplicity of interests. This symposium presents four of these studies to convey the current state of Soar as a UTC (their names are marked with asterisks in the figure). This short paper provides additional breadth and context.

THE SOAR ARCHITECTURE

We review here the basic structure of the Soar architecture, which has been described in detail elsewhere [8, 13, 20]. Soar formulates all tasks in *problem spaces*, in which *operators* are selectively applied to the *current state* to attain *desired states*. Problem spaces appear as triangles in Figure 2 (which describes a Soar system for comprehending natural language). Problem solving proceeds in a sequence of *decision cycles* that select problem spaces, states, and operators. Each decision cycle accumulates knowledge from a long term *recognition memory* (realized as a production system). This memory continually matches against *working memory*, elaborating the current state and retrieving preferences that encode knowledge about the next step to take. Access of recognition memory is involuntary, parallel, and rapid (assumed to take on the order of 10 milliseconds). The decision cycle accesses recognition memory repeatedly to quiescence, so each decision cycle takes on the order of 100 milliseconds.

If Soar does not know how to proceed in a problem space, an *impasse* occurs. Soar responds to an impasse by creating a subgoal in which a new problem space can be used to acquire the needed knowledge. If a lack of knowledge prevents progress in the new space, another subgoal is created and so on, creating a goal-subgoal hierarchy. Figure 2 shows how *multiple problem spaces* arise. Once an impasse is resolved by problem solving, the *chunking* mechanism adds new productions to recognition memory encoding the results of the problem solving, so the impasse is avoided in the future. All incoming perception and outgoing motor commands flow through the state in the top problem space (which occurs above the spaces in Figure 2).

FOUR EXAMPLES OF RECENT PROGRESS

NL-Soar (Huffman, Lehman, Lewis and Tessler)

The goal of the NL-Soar work is to develop a general natural language capability that satisfies the constraint of real-time comprehension. To achieve rates of 200-300 words per minute, NL-Soar must *recognitionally* bring to bear multiple sources of knowledge (e.g., syntactic, semantic, and task knowledge). In addition to

[1] Soar research encompasses artificial intelligence and human computer interaction (HCI) as well, which we will largely ignore here.

[2] We include several unpublished studies to better convey Soar's current state.

Time Scale	Name	Domain	Phenomena modelled	Notes	References
short	Neuro-Soar	Neural net implementation of Soar	Implementation technology of the architecture	A	[17]
	Visual Attention	Attention within field of view	Conjunctive, disjunctive feature searches; illusory conjunction	A	[27]
	Immed. Behavior	Simple reaction tasks; typing	Performance times	Z	[13]
	ET-Soar	Intelligent tutor for electrostatics	Perception-action loop in problem-solving with external display	S	[26]
	Phone Operator	Operators assisting in collect calls	Performance time of task involving multiple modalities	Z	[4]
	* Browser-Soar	Computer browsers	Browsing an on-line reference manual	P	[5]
	* Robo-Soar	Puma robot arm in blocks world	On-demand flexible planning, interruptibility	S	[7,9]
	* Hero-Soar	Hero mobile robot picking up cups	Reactive execution; operating in noisy, uncertain environments	S	[9]
	* NL-Soar	Natural language comprehension	Real-time comprehension; integrates multiple knowledge sources	S	[11]
	IR-Soar	Immediate reasoning tasks (< 60s)	Regularities across multiple IR tasks (syllogisms, Wason, etc.)	Q	[15,16]
	IR-Soar(Syl)	Categorical syllogisms	Individual differences	I	[15,16]
	BI-Soar	Taking instructions for IR tasks	Acquisition of new tasks from natural language instructions	S	[11]
	* SC-Soar	Series completion	Relative difficulty of different series	Q	
	Subtraction	Multi-column subtraction	Learning new subtraction procedure; two common bugs	S, N	[20]
	Cryptarithmetic	Cryptarithmetic puzzles	Control of search in cryptarithmetic task space	P, N	[13]
	TOH-Soar	Tower of Hanoi	Strategy change, within-trial learning	P	[21]
	Verbal Learning	Paired-associate, cued recall, etc.	Learning new declarative knowledge (knowledge level learning)	S	[19]
	Inductive KLL	Concept learning	Inducing concepts from examples (knowledge level learning)	S	[18]
long	Text Editor	Learning to use a text editor	Conceptual errors in using a new text editor	Q	[28]
	Data-Soar	Learning new data structures	Use of analogy in learning "stack" data type	P	[24]
	Generic Design	Architectural, instructional design	Design of instructional material for word processor	P	[14]
	Lexical Acquis.	Early lexical acquisition	Prototype effects; underextension; one-concept-one-word bias	Q	[12]
	ABC-Soar	Analysis-based concept acquisition	Acquisition of number conservation concepts	S	[23]
	Balance Beam	Balance beam task	Developmental transitions in balance beam task	Q, N	[13]

NOTES: A: work on the architecture itself I: accounts for individual differences N: work no longer in progress P: protocol simulation
Q: qualitative coverage of phenomena S: sufficiency demonstration Z: zero parameter predictions

Figure 1: **Some efforts on Soar as a unified theory of cognition.**

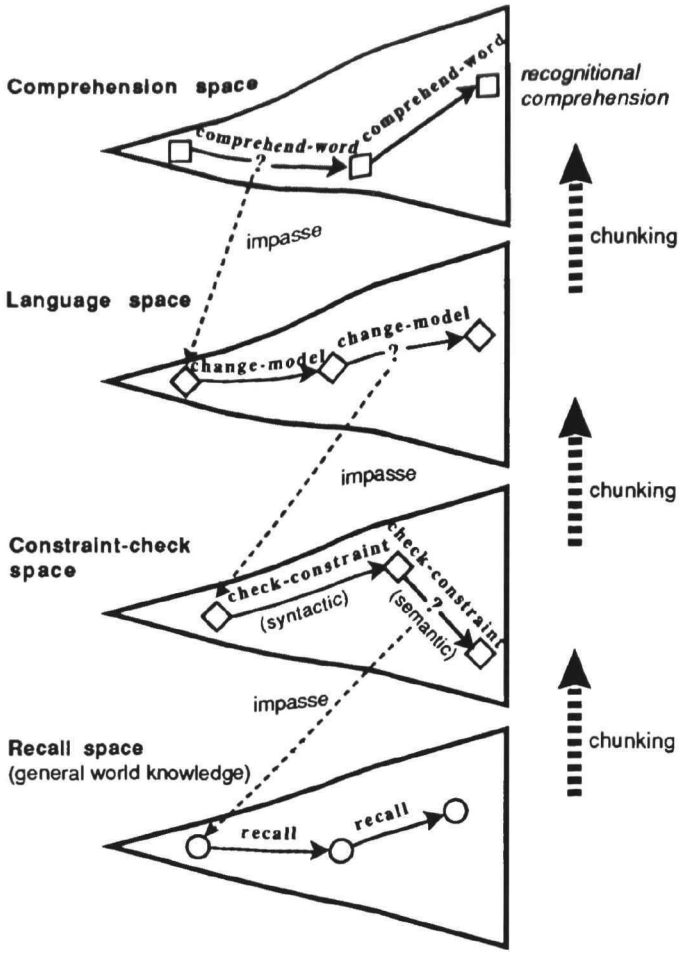


Figure 2: **Achieving recognitional comprehension from deliberate comprehension in NL-Soar.**

supporting this kind of processing, the system must integrate with the rest of Soar to work in any task that requires language understanding.

NL-Soar realizes real-time comprehension by applying a *single operator* to comprehend each word as it is read, dropping into lower problem spaces to complete the comprehension when the required knowledge is not directly available (i.e., an impasse is reached). Figure 2 shows this behavior. As in previous work [11], we assume that comprehension builds a *situation model* of what the utterance is about, and an *utterance model* that encodes the syntactic structure of the utterance. When Soar impasses on a **comprehend-word** operator, it drops into the **Language** space, which has operators that make changes to the two models. Any kind of knowledge can propose an operator (e.g., syntactic or semantic), but operators have constraints that must be satisfied before they can apply. These constraints can be arbitrarily complex syntactic, semantic, or pragmatic tests. The constraints are checked serially in the **Constraint-check** space, which consults the situation and utterance models. Checking semantic constraints may require accessing general world knowledge, which occurs in the **Recall** space. Other spaces may be involved in comprehension; for example, referent resolution happens in the **Resolve-referent** space (not shown in the figure). Once the deliberate problem solving produces the knowledge to resolve the **comprehend-word** operator's impasse at the top, chunking adds new elements to long term memory so comprehension can proceed by recognition for similar utterances and contexts in the future. These chunks do the constraint checking in parallel, integrating into a single recognition the disparate knowledge sources tapped in the deliberate problem solving.

NL-Soar is a functioning language system: its syntactic knowledge currently comprises about 75% of James Allen's outline of English in [1], and it can handle the semantics of simple instructions for the immediate reasoning [11] and Robo-Soar blocks-world tasks [7, 9]. The novel aspect of the recent work is using chunking to routinely move knowledge up the problem space hierarchy to create an integrated recognitional comprehension system. A preliminary analysis over a very small set of test sentences indicates that about half the chunks perform lexical access (i.e., the chunks are specific to lexical items). The remaining non-lexical chunks are also fairly specific, usually tied to particular syntactic forms and semantic categories. While there is transfer to new utterances, the transfer is limited by the specificity of the chunks. This implies a model of comprehension that depends on a large number of fairly specific chunks [13]. This does not mean there is no generality in NL-Soar's linguistic knowledge. Quite the opposite, the **Language** space comprises very general sources of knowledge. It simply means that the generality is limited at the recognitional level. Additional research is necessary to fully establish the sufficiency of this model.

SC-Soar (Simon)

The goal of the SC-Soar work is to develop a model of regularity detection as it occurs in series completion tasks (e.g., A, B, A, C, A, ?). Solving these puzzles requires the capability to hypothesize and test the underlying rule of the series. The series-completion task provides a microcosm for studying concept acquisition; it involves inducing a concept from examples and encoding the environment in terms of the new concept.

SC-Soar solves a series completion problem by casting it as a comprehension task: comprehension operators are applied to each item in the series, and subspaces encode the knowledge of how to hypothesize and test the underlying rule. The structure is that of NL-Soar, although the knowledge is not about language but about the relations that characterize the series, and the focus is on the deliberate phase of this specialized comprehension. Comprehending an item involves finding relationships between the item and other items in the series (relationships can be alphabetic or identity). To establish a relationship, attention must be focused on at least one other item; the *attention operators* posited for the work on immediate reasoning [16] provide this functionality. Attending to other items may also trigger knowledge to hypothesize a period size for the series. Once a period is hypothesized, SC-Soar *re-represents* the series by structuring it into groups of items. This is not accomplished in one massive step, but occurs by re-reading the series. Thus, SC-Soar exhibits a form of progressive deepening. Comprehension chunks learned during the first pass transfer to subsequent passes, so that some of the deliberate processing can be avoided. After this re-encoding, comprehension proceeds by establishing relationships between groups rather than single items. A relationship between groups, combined with the simpler relationships within groups, establishes a hypothesis for the underlying rule of the series. This hypothesis is tested by generating expectations and matching them against the rest of the series.

The novel features of SC-Soar are structuring the system into comprehension operators, modulating problem solving by attention, and learning during the task. The first two have independent support for their plausibility from other work [11, 13, 16]. The learning is important for modeling progressive deepening behavior and improvement across trials. Currently SC-Soar solves ten out of fifteen series in the original Simon and Kotovsky set [22]. The five series it cannot solve were among the most difficult. For the ten it solves, the number of decision cycles it takes to hypothesize the correct pattern provides a measure of duration that corresponds closely to the relative difficulty of these series for the subjects. The learning has not yet been compared with human data.

Robo-Soar and Hero-Soar (Laird)

The goal of the Robo-Soar and Hero-Soar work is to get Soar to interact with an external environment. The functional demands of working in a dynamic environment include *reactive execution*, *interruptibility*, *flexible on-demand planning*, and the *conversion of deliberate planning to reactive execution*. Robo-Soar and Hero-Soar are robotic systems developed to explore how Soar can support these capabilities [7, 9]. Robo-Soar controls a Puma arm and receives perceptual input through a camera; its task is to align blocks in a work area, unless interrupted by a flashing "trouble" light, in which case it should immediately push a button. Hero-Soar controls a mobile robot with grippers and sonar sensors; its task is to navigate around a room looking for cups to retrieve.

Robo-Soar and Hero-Soar cope with the demands of external interaction by taking advantage of the three different ways that Soar can intend actions: by recognition, by deliberate operator selection, and by unrestricted problem solving. At the lowest level, actions can be evoked by a single chunk in recognition memory. This supports high-speed response for automatic reflexes, such as stopping the wheel motors when Hero-Soar detects an object in its path. No deliberation is involved, hence the action cannot be modulated by additional knowledge. At the next level of control, Soar chooses an action by selecting an operator in a problem space. For example, in Robo-Soar operators exist for **close-gripper**, **open-gripper**, and **move-gripper**. At each decision cycle, recognition memory retrieves all the relevant preferences about what action to do next, and the decision procedure interprets these preferences to make a unique selection. Thus, the decision is based upon an integration of all immediately available knowledge, rather than an isolated stimulus. This makes Soar *interruptible*, as at each decision cycle any operator can be proposed and considered (e.g., the operator to push the button). The cost of bringing more knowledge to bear is time. However, the decision cycle is fast enough (about ten decisions per second) that the operator level effects reactive execution. When Soar lacks the knowledge to select the next action uniquely, an impasse arises. Soar responds by formulating this selection as a problem to be solved in a problem space. Planning is but one of a number of methods available. For example, Soar could ask an outside agent for help. Once the impasse is resolved, chunks are added to recognition memory that will allow the decision to be made in the future without problem solving. Thus, as in NL-Soar (Figure 2), chunking supports the conversion of deliberate processing to reactive execution.

Robo-Soar and Hero-Soar are not psychological models of perception and action. However, whether the goal is to build a mobile robot or model human behavior, the functional demands of working in a dynamic, uncertain environment are the same. From the UTC perspective, the Robo/Hero-Soar work establishes that Soar is sufficient to handle some of these demands and permits the exploration of how perception and motor behavior integrate with central cognition and learning. It paves the way for adopting psychologically realistic perceptual and motor systems.

Browser-Soar (John)

The goal of the Browser-Soar work is to model how humans interact with a reactive environment (a computer browser) in a rapid perception-action loop [5]. Browser-Soar is intended to be a detailed psychological model of the moment-by-moment interaction.[3] Browsers are application programs that allow multiple access paths to data bases via recognition of pointers (as opposed to retrieval by a query command language). The use of browsers is characterized by both deliberate search and opportunistic switches in search strategy. The specific browser studied is the on-line help system for cT, an interactive graphics programming language[4]. This browser consists of three windows: one provides access to topics via a hierarchical menu, another

[3]Browser-Soar does not yet interface with actual robotic effectors and perceptual devices.

[4]cT was developed at the Center for the Design of Educational Computing (CDEC), Carnegie Mellon University.

provides access via an alphabetic list, and the third displays the help text for a selected topic.

Browser-Soar consists of a set of problem spaces that provide the capability to search deliberately through the help windows, while allowing recognition of new items to trigger knowledge at any time that may change the search strategy. The top problem space in Browser-Soar (**Browse**) is entered when an impasse arises in the task space for programming in the cT language. A browsing episode involves bringing up the help window, finding the appropriate help, and applying the newly found information to the problem at hand. Each of these activities corresponds to an operator in the **Browse** space. Currently, Browser-Soar implements the **find-appropriate-help** operator. Applying this operator results in an impasse because the operator cannot be implemented by recognition. Soar responds by setting up another problem space, with operators that define the search criteria (e.g., what labels to look for in the help windows), define the evaluation criteria (how to decide that some piece of information will actually help resolve the impasse in the task space), carry out the search, and evaluate the search results. Each of these operators is also implemented in a problem space; for example, carrying out the search for the defined criteria is accomplished in a space with operators that select among search methods and execute them. At the bottom of the problem space hierarchy are motor operators that control mouse and keyboard actions, and cognitive operators that can be applied with directly available knowledge. The operators in Browser-Soar can be viewed as deliberate goals, and this organization is useful for modeling the goal-oriented component of browsing as well as the mechanics of manipulating the windows used for browsing. Data-driven, opportunistic behavior emerges because an additional operator, **evaluate-new-items**, is proposed whenever new information is brought within the scope of attention (**evaluate-new-items** is available in every Browser-Soar problem space). The current problem solving is thus *interrupted* (as in Robo/Hero-Soar) so the new items may be considered, possibly suggesting a more relevant path to pursue.

Browser-Soar simulates in detail a 67 second episode of a cT programmer using the on-line help to try to find a particular graphics command. The simulation shows the user's behavior to be largely GOMS-like [2], i.e. a *routine* cognitive skill, even though, because of the continuous rapid feedback, the behavior might seem more open and intelligently spontaneous. On the other hand, the episode contains three failed accesses of help text and two changes of search strategy. To handle these requires forms of control that lie outside of the GOMS-style procedural hierarchy (e.g., the interrupt). Thus, Browser-Soar not only shows that this browsing behavior is GOMS-like, but characterizes the ways in which it departs from that simple model.

MAKING PROGRESS TOWARD A UNIFIED THEORY OF COGNITION

We now step back from an examination of specific models to look at how Soar as a whole is developing. Breadth should be the hallmark of a UTC. In fact, a UTC effort that engages a large number of independent investigators will by nature proceed breadth-first. Soar certainly fits this model, and continues to cover a wide range of cognitive phenomena (Figure 1)[5]. The task variety exhibited in Figure 1 ranges from traditional problem solving tasks like Tower of Hanoi, to highly skilled tasks such as natural language, to planning and interacting in a dynamic external environment. The span of time-scales is also evident from Figure 1, ranging from immediate response tasks to developmental tasks. We are beginning to take advantage of *cross-domain integration*. By integrating different capabilities to model a whole task, we can bring more constraints to bear and reduce theoretical degrees of freedom. This kind of integration is evident in the close relation between NL-Soar and the series completion model (SC-Soar), and in work on taking

[5]We talk about individual attempts to use Soar as a cognitive theory as if they were separate systems, and give them unique names, such as XX-Soar. In fact, each such system is the same Soar, but with different knowledge. Such knowledge is often ad-hoc, which poses a problem of identifying the contributions of the theory, but this is a problem for all of psychology. See [11] for an approach to this problem with Soar.

instructions (NL-Soar, BI-Soar, and IR-Soar [11]).

In addition to achieving wide coverage, Soar must ultimately provide theories that press the scientific frontiers of specific domains. Soar is beginning to achieve such depth in diverse ways. The immediate behavior models provide *zero-parameter predictions* of performance times for simple reaction, choice reaction, transcription typing, and stimulus-response compatibility tasks [4, 13]. IR-Soar(Syl), the syllogistic reasoning model [15, 16], has recently been extended to give a detailed account of *individual differences*. The Browser-Soar, Data-Soar [24], Generic Design [14], and TOH-Soar [21] models provide *simulations of thinking aloud protocols* for their respective tasks. The Tower of Hanoi simulation is particularly interesting as it uses chunking to model strategy change. Figure 1 notes some of the different ways that Soar theories model behavior: accounting for individual differences (I), simulating protocols (P), providing qualitative coverage of phenomena (Q), demonstrating the sufficiency of Soar to accomplish the task (S), and making zero parameter predictions (Z).

Soar theories can be novel in their predictions and explanations, and/or synthetic, by incorporating existing theories and well understood mechanisms. For example, besides providing a new theory of individual differences, IR-Soar(Syl) is synthetic in its incorporation of the theory of mental models [6]. Browser-Soar is essentially a GOMS-like model, but introduces new control mechanisms. NL-Soar is synthetic in its assimilation of existing linguistic knowledge, but is novel in its method for automatically becoming recognitional. In contrast, the immediate behavior theory builds mostly on previous work in HCI [2, 3], while the model of the balance beam task [13] is completely unique to Soar.

FUTURE DIRECTIONS

We believe it is now fair to assess Soar as a genuine candidate for becoming a generally useful UTC. Soar's breadth is impressive and there are beginning to be places where it makes specific contributions at the frontier. However, the coverage remains quite patchy, which substantially diminishes the assessment of Soar as a coherent theory. Also, Soar efforts often evolve along idiosyncratic paths, viewed against the standard ways theory and data develop in psychology. For instance, NL-Soar's mechanism for the transition from deliberate to recognitional comprehension emerges well before NL-Soar confronts any standard linguistic or psycholinguistic phenomena. There are reasons for this priority, namely, the functional necessity of creating and growing the mass of **comprehend-word** operators. But it leaves NL-Soar deficient when evaluated by current criteria in either linguistics or psycholinguistics. More examples can be easily found. Thus, our net assessment is that Soar must still be viewed as nascent, and a substantial wait-and-see attitude is still justified.

Yet if one looks for encouraging results, they are also easily discerned. Integration of perception, cognition, and motor behavior is proceeding, although much still needs to be done. The little cluster of integration of language comprehension (NL-Soar), immediate reasoning (IR-Soar), instruction-taking (BI-Soar), and concept formation (SC-Soar) is an important harbinger.

Important changes are occurring that involve evolution of the Soar architecture itself. We have shifted to an underlying processing of states by continuous modification rather than discrete copying [10]. We continue to move toward using an *annotated models* representation [16] in all of our systems. The effort is already underway to extend Soar perception to model visual attention [27]. We have become convinced that the existing recognition memory is computationally unrealistic, so work is in progress to examine alternative formulations [25]. We are also exploring the possibility of implementing the Soar architecture using a neural network [17].

Acknowledgements

Many thanks to Thad Polk and Kathryn Swedlow for useful comments. This work was supported by the Information Sciences Division, Office of Naval Research, under contract N00014-86-K-0678, and by the National Science Foundation fellowship program in which Richard Lewis participates. The views expressed in this paper are those of the authors and do not necessarily reflect those of the supporting agencies. Reproduction in whole or in part is permitted for any purpose of the United States government. Approved for public release; distribution unlimited.

References

[1] Allen, J. **Natural Language Understanding**. Benjamin/Cummings, Menlo Park, CA, 1987.

[2] Card, S., Moran, T. P., and Newell, A. **The Psychology of Human-Computer Interaction**. Erlbaum, Hillsdale, NJ, 1983.

[3] John, B. E. *Contributions to Engineering Models of Human-Computer Engineering*. Department of Psychology, Carnegie Mellon University, May 1988. *PhD. Thesis.*

[4] John, B. E. *Extensions of GOMS analyses to expert performance requiring perception of dynamic visual and auditory information.* To appear in the *Proceedings of the Conference on Human Factors and Computing Systems*, April, 1990.

[5] John, B. E., Newell, A., and Card, S. *Browser-Soar: A GOMS-like model of a highly interactive task. Talk presented as the Human Computer Interaction Consortium Winter Workshop, San Diego, CA, February 12, 1990.*

[6] Johnson-Laird, P. **Mental Models**. Harvard, Cambridge, MA, 1983.

[7] Laird, J. E., Hucka, M., Yager, E. S., and Tuck, C. M. *Correcting and extending domain knowledge using outside guidance.* in: **Proceedings of the Seventh International Conference on Machine Learning**. 1990.

[8] Laird, J. E., Newell, A., and Rosenbloom, P. S. *Soar: An architecture for general intelligence.* **Artificial Intelligence**, vol. 33 (1987), pp. 1–64.

[9] Laird, J. E. and Rosenbloom, P. S. *Integrating planning, execution, and learning in Soar for external environments.* February 1990. *Artificial Intelligence Laboratory, The University of Michigan. An earlier version of this paper will appear in the 1990 AAAI Spring Symposium Workshop on Planning in Uncertain, Unpredictable, and Changing Environments.*

[10] Laird, J. E., Swedlow, K., Altmann, E., and Congdon, C. B. *Soar 5 User's Manual.* 1990. *In preparation.*

[11] Lewis, R. L., Newell, A., and Polk, T. A. *Toward a Soar theory of taking instructions for immediate reasoning tasks.* in: **Proceedings of the Eleventh Annual Conference of the Cognitive Science Society**. 1989, pp. 514–521.

[12] Miller, C. S. and Laird, J. E. *A Simple, Symbolic Model for Associative Learning and Retrieval.* March 1990. *Artificial Intelligence Laboratory, The University of Michigan. Unpublished.*

[13] Newell, A. **Unified Theories of Cognition**. Harvard University Press, Cambridge, Massachusetts, 1990. *In press.*

[14] Pirolli, P. *Generic design. Talk presented at the Seventh Soar Workshop, Carnegie Mellon University, Pittsburgh, PA, February 23-25, 1990.*

[15] Polk, T. A. and Newell, A. *Modeling human syllogistic reasoning in Soar.* in: **Proceedings of the Tenth Annual Conference of the Cognitive Science Society**. 1988, pp. 181–187.

[16] Polk, T. A., Newell, A., and Lewis, R. L. *Toward a unified theory of immediate reasoning in Soar.* in: **Proceedings of the Eleventh Annual Conference of the Cognitive Science Society**. 1989, pp. 506–513.

[17] Rosenbloom, P. S. *A symbolic goal-oriented perspective on connectionism and Soar.* in: **Connectionism in Perspective**, edited by R. Pfeifer, Z. Schreter, F. Fogelman-Soulie, and L. Steels. Elsevier (North-Holland), Amsterdam, The Netherlands, 1989, pp. 245–263.

[18] Rosenbloom, P. S. and Aasman, J. *Knowledge Level and Inductive Uses of Chunking (EBL).* February 1990. *Information Sciences Institute, University of Southern California. Unpublished.*

[19] Rosenbloom, P. S., Laird, J. E., and Newell, A. *The chunking of skill and knowledge.* in: **Working Models of Human Perception**, edited by H. Bouma and A. Elsendoorn. Academic Press, London, England, 1988, pp. 391–410.

[20] Rosenbloom, P. S., Laird, J. E., Newell, A., and McCarl, R. *A preliminary analysis of the Soar architecture as a basis for general intelligence.* in: **Proceedings of the Workshop on Foundations of Artificial Intelligence**, edited by D. Kirsh and C. Hewitt. MIT Press, Cambridge, Massachusetts, 1989.

[21] Ruiz, D. and Newell, A. *Tower-noticing triggers strategy-change in the Tower of Hanoi: A Soar model.* in: **Proceedings of the Eleventh Annual Conference of the Cognitive Science Society**. 1989, pp. 522–529.

[22] Simon, H. A. and Kotovsky, K. *Human acquisition of concepts for sequential patterns.* **Psychological Review**, vol. 70 (1963), pp. 534–546.

[23] Simon, T. *Modeling conceptual development: A computational account of conservation learning.* in: **Computational Approaches to Concept Formation**, edited by D. Fisher and M. Pazanni. Morgan Kaufmann Publishers, Inc., Los Altos, California, 1990. *In press.*

[24] Steier, D. and Adelson, B. *Data-Soar: Simulating a student learning about data structures. Talk presented at the Seventh Soar Workshop, Carnegie Mellon University, Pittsburgh, PA, February 23-25, 1990.*

[25] Tambe, M. and Rosenbloom, P. S. *Eliminating expensive chunks by restricting expressiveness.* in: **Proceedings of the Eleventh International Joint Conference on Artificial Intelligence**. 1989, pp. 731–737.

[26] Ward, B. *A Soar-Based Intelligent Tutor for Electrostatics.* May 1988. *Thesis proposal, Computer Science Department, Carnegie Mellon University.*

[27] Wiesmeyer, M. and Laird, J. E. *A Computer Model of 2D Visual Attention.* March 1990. *Artificial Intelligence Laboratory, The University of Michigan. Unpublished.*

[28] Young, R. M. and Whittington, J. *Using a knowledge analysis to predict conceptual errors in text-editor usage.* To appear in the *Proceedings of the Conference on Human Factors and Computing Systems*, April, 1990.

Symposium: What is Cognitive Neuroscience?

Participants: David Caplan (Harvard Medical School), Michael S. Gazzaniga (Dartmouth Medical School), Stephen M. Kosslyn (Harvard University), Michael I. Posner (University of Oregon), Larry Squire (UCSD).

The purpose of this symposium is to characterize the emerging field of cognitive neuroscience. The symposium will depart from the usual format in which participants deliver formal presentations. Instead, each participant will briefly describe what he believes to be a paradigm example of research in cognitive neuroscience, and these presentations will be the impetus for a round table discussion. The discussion will focus on key issues in cognitive neuroscience, including the roles of animal models, cognitive psychology, computational models, neurology, and various formalisms.

What's New In Language Acquisition?

Chairs: Steven Pinker & Kenneth Wexler, MIT

Linguistic facts that our parents never taught us. *Stephen Crain, Department of Linguistics, University of Connecticut.* I present the findings of several studies of young children's knowledge of grammatical phenomena for which there is no corresponding input from the environment. The findings are interpreted as support for the principles of Universal Grammar.

Feature Blindness: An Inherited Disorder of Grammar. *Myrna Gopnik, Department of Linguistics, McGill University.* I discuss a syndrome of inherited developmental dysphasia that selectively impairs the productive use of morphological features, and that appears to be controlled by a single dominant gene.

Why do children overregularize? *Steven Pinker, Department of Brain and Cognitive Sciences, MIT; Alan Prince, Program in Linguistics and Cognitive Science, Brandeis University; Michelle Hollander, John Kim, Gary Marcus, Sandeep Prasada, Michael Ullman, Department of Brain and Cognitive Sciences, MIT.* Why do children make errors like *breaked*? Why do they get worse as they get older? How often do they make them? Why do they make more errors with some verbs than others? When do they get better, and why? We present new data on all these questions, and discuss their relevance to the debate about rule-based and associationist theories of language acquisition.

The Development of Structural Relations in Child Language: Binding and Types of Movement. *Kenneth Wexler, Department of Brain and Cognitive Sciences, MIT.* Results are presented of experiments which show young children's knowledge of the abstract principles of structural binding and movement. Questions relating to the learning of language-specific properties and of the maturation of linguistic principles are discussed.

Attracting Attention

Participants and Paper Titles

Patrick Cavanagh (Harvard) - Pursuing moving objects with attention

Ken Nakayama (Harvard) - Visual search and its relation to depth cues

Jeremy Wolfe (MIT) -Three Aspects of the Parallel Guidance of Visual Attention

Steven Yantis (Johns Hopkins) - Attentional Priority in Vision

Chair

Anne Treisman (Berkeley)

Introduction

The early stages of visual processing appear to operate in parallel across the entire visual field. There are retinotopic arrays of neurons that appear to support the processing of basic features such as size, color, orientation, and so forth (VanEssan and Maunsell, 1983) but not of more elaborate properties (e.g. conjunctions of two basic features - Treisman and Gelade, 1980; Wolfe, Cave, and Franzel; 1989; Treisman and Sato, 1990). Parallel processing is not feasible for all visual tasks. The brain is simply not large enough (e.g. Tsotsos, 1990). The parallel stage seems to be restricted to a few simple operations on a few basic features (e.g. Wolfe et al, 1990). For more complex visual tasks, the ability to perform these tasks is restricted to a portion of the visual field. For many tasks, the relevant portion of field is not fixed but can be selected by attention. Thus, the large-scale architecture of the visual system seems to consist of a parallel front end leading to a processing bottleneck. After the bottleneck, the field of action of various limited capacity processes can be moved about under attentional control. Experimental evidence suggests that those movements can occur at a rate of 15-25 movements per second (e.g. Sagi and Julesz, 1986) though this rate may be much slower if attention becomes "engaged" at one locus (Mackeben and Nakayama, 1988).

For real-world visual tasks, it is not practical to allow visual attention to wander about the visual field at random. To use this architecture efficiently, the parallel stage should guide the deployment of attention. In this fashion, attention could focus limited capacity processes where they are needed and not waste time on sophisticated processing of the wrong part of the visual field.

While the logic of having a high capacity, parallel stage guide subsequent, limited capacity processes seems clear, it is not entirely clear how that guidance is provided. As noted above, the parallel stage has very limited capabilities. In this symposium we intend to discuss the nature of those limitations and the mechanics of attracting attention.

Pursuing moving objects with attention.

Patrick Cavanagh
Dept. of Psychology, Harvard U, Cambridge, MA 02138

When an observer fixates a point in space but attends to a moving object, there is a clear perception of motion. There are two possible sources for that perception. It may be due to the selection of the low-level motion signals generated by the object or it may be derived from the signals generated by the attention system in order to track the object. This second possibility is very much like the efferent copy model for the perception of target motion during pursuit eye movements. The target does not move on the retina during the tracking eye movements but it is nevertheless perceived to move in space. According to the efferent copy model, this motion percept is derived from the eye movement signals required to track the target.

To study whether the perceived motion of objects tracked without eye movements is derived from a similar but covert signal, tracking experiments were run using counterphase gratings. A counterphase grating is the sum of two identical gratings moving in opposite directions, left and right, for example. The stimulus contains no net motion but if it is tracked with eye movements in either direction, a moving grating is seen. To eliminate tracking eye movements, a radial, counterphase grating was constructed within a circular annulus. A radial grating rotating clockwise was added to a radial grating rotating counterclockwise, producing a counterphasing grating.

When an observer fixated the center of the radial grating, no net motion was evident. However, if the observer attended to individual spokes (while still fixating the center), a clear impression of motion was produced in the direction of the attentional tracking and this direction could then be changed at will. Observers most often reported attending to a pair of symmetrically opposed spokes in the grating and following their motion around the annulus. While attending to a given pair of spokes, the remaining areas of the annulus did not appear to rotate clearly in either direction. With practice, attention could be directed to other patterns of spokes such as a triple-pointed "Mercedes-Benz" symbol or a four-point cross. The attended regions in these patterns of spokes were not contiguous, but all the spokes in the attended pattern rotated in the same direction.

These results demonstrate that attention can track moving elements of an ambiguous display and disambiguate one direction of motion in a compound stimulus. The ability to attend to different groupings of elements (pairs, three-pointed and four-pointed stars) suggests a higher-order motion process that uses simple forms as place keepers to group separate elements into a unit that can be tracked over time. This process may be related to the place-keeping process that Pylyshyn and Storm studied with fields of independently moving luminance disks.

A second experiment showed that the motion perceived during attentional tracking is based on a higher-order motion analysis that differs greatly from low-level motion in its relative sensitivity to luminance and color. A clockwise grating defined by luminance was added to a counterclockwise grating defined by color. The contrast of the luminance grating was adjusted until neither direction of rotation dominated when fixating the center of the annulus. At this point, a small increase in the contrast of the luminance grating produced an impression of global rotation in the direction of the luminance grating(clockwise) whereas a small decrease produced an impression of global rotation in the direction of the color grating (counterclockwise). With the two directions of motion balanced and eyes fixated at

the center, observers could easily track the counterclockwise motion of the color grating when asked to attend to individual spokes. However, they could not track the luminance spokes rotating in the opposite direction. In fact, very little contrast (about 10%) was required in the luminance grating to reach the point of motion balance. In the presence of the color patterns, the 10% contrast luminance patterns were difficult to see and it is reasonable to assume that the ease of tracking pattern elements depends on the visibility of the patterns. On the other hand, visibility is not such an important factor for the low-level motion system. The results here, as well as previous results, show that a luminance stimulus produces a strong low-level motion signal even when it is barely visible whereas a color stimulus produces only a weak low-level motion signal even though it is highly visible.

When the low-level motion signals from the color and luminance gratings were balanced in the stimulus, no net motion was seen in either direction without attention to individual elements. Selection of low-level signals can therefore be ruled out as the source of the perception of motion during tracking because, based on low-level signals, it should be as easy to track the motion of the luminance elements as that of the color elements. The results therefore suggest that the impressions of motion during tracking must be derived from the signals that displace the tracking window. Indeed, although the rotating color gratings appeared to slow down or even stop when the colors were equiluminous, the motion of the color elements re-emerged when they were being tracked.

Visual search and its relation to depth cues

Ken Nakayama
Dept. of Psychology, Harvard U, Cambridge, MA 02138

Some conjunctive searches can be done in parallel. That is, there is no increase in reaction time as the number of items is increased. Other conjunctive searches (using different visual dimensions), however, are more difficult and reaction time increases for increasing set sizes. Our working hypothesis is that feature grouping among some visual dimensions is stronger than in others. In particular, those features which reliably co-vary with the distance of surfaces in real world scenes seem most likely to support grouping. As such, there is the tendency for parallel conjunctive search to be most prominent for tasks employing the dimensions of binocular disparity, size, spatial frequency, and motion as opposed to color and orientation/form.

In a new series of experiments designed to evaluate the nature of perceptual grouping in visual search tasks, Mary Bravo and I have shown very different findings depending on whether the observer has knowledge of the target color. We find that when the color of a target is unchanged from trial to trial (the observer is searching for a known target), performance is unaffected by the number of distractors. When the color of a target is changed from trial to trial (the observer is searching for an unknown target), reaction times decrease with increasing numbers of distractors. The importance of distractor number in the search for an unknown target is even greater when target and distractors are randomly distributed across two disparity planes.

Three Aspects of the Parallel Guidance of Visual Attention

Jeremy M Wolfe
Dept. of Brain and Cognitive Sciences, MIT, Cambridge, MA 02139

As noted in the Introduction, visual processing can be crudely divided into two parts: A "front-end" capable of processing a few simple features in parallel across the visual field and a "back-end" consisting of more powerful processes that can operate over only a restricted portion of the visual field. These processes can be directed to different parts of the field by attention. To move the limited capacity processes in a sensible and efficient manner, attention should be *guided* by information collected in parallel. The Guided Search model (Wolfe, Cave, and Franzel, 1989; Cave and Wolfe, 1990) is an effort to understand that guidance in the context of visual search. In a visual search task, a subject looks for a **target** item (or items) among a variable number of **distractor** items. Various measures of performance may be used (e.g. Klein and Farrell, 1989; Bergen and Julesz, 1983). In our experiments, we follow Treisman (1988) in measuring reaction time (RT) as a function of the total number of items presented. Some search tasks yield highly efficient searches, independent or nearly independent of set size. The clearest examples are searches for a target defined by a simple feature (color, size, etc) presented among homogeneous distractors (e.g. a red target among green distractors) (Treisman and Souther, 1985; Duncan and Humphreys, 1989).

When simple feature information is of no use, RTs increase linearly with set size. The increase is twice as great for target-absent trials as for target-present trials. These results are consistent with a self-terminating search governed by serial movements of attention at a rate of 40-60 msec/item. (e.g. Sagi and Julesz, 1986). A search for a "T" among "L"s is an example of such a search. If the Ts and Ls can rotate, they can be distinguished only by determining the relative position of the two line segments. This task appears to require serial deployment of attention.

From the vantage point of Guided Search, the most interesting cases are those where simple feature information is present even if no single feature is adequate to differentiate target and distractors. An example is the search for a conjunction of two features. If the target is a red vertical item while the distractors are red horizontal and green vertical, no single feature can be used to discriminate targets from distractors. However, if attention can be guided toward "red" items and toward "vertical" items, the combination of these two sources of guidance could direct attention toward "red vertical" items. Guided search for a conjunction of two basic features can be much more efficient than a search where no guidance is possible. (Wolfe, Cave, and Franzel, 1989; Egeth, Virzi, and Garbart, 1984; Nakayama and Silverman, 1986; Treisman and Sato, 1990).

In the Guided Search model, attention is guided by the activation in the parallel stage. There are two components to parallel stage activation. Bottom-up activation is an accelerating, non-linear function of the mean of the differences between an item and other items in its neighborhood. Top-down activation is a function of the similarity between an item and the internal representation of the target. These activations are computed independently with each parallel feature module. Thus, in a search for a red vertical item, there would be relevant bottom-up activations from both color and orientation modules based on local variations in those features. Further, there would be independent top-down activation in the same modules based on the similarity of each item to "red" for the color module and "vertical" for the orientation module. All activations are summed and perturbed by noise.

Attention moves from location to location on the basis of total activation starting with the most active and continuing until the activation falls below some threshold. (See Cave and Wolfe, 1990, for details)

In this paper, we will discuss three aspects of the parallel guidance of attention:

1) Bottom-up activation can be based on local contrast, independent of stimulus identity. Consider the following task (Each pattern stands for a different color).



POSSIBLE TARGETS POSSIBLE DISTRACTORS

It is easy to find these targets because they contain a high chromatic contrast edge. It is not necessary for target colors to be unique or known. A simple high degree of local variation attracts attention.

2) Top-down activation is based on perceptual "categories". Top-down activation is based on the similarity between an item and an internal representation of target properties. Evidence from searches for a target orientation among two or more distractor orientations suggests that this internal representation is categorical (and not, for example, geometrical). Attention can be directed toward "steep" (or "shallow") items or toward "left" (or "right") tilted items but not toward targets specified by angle of orientation or even by the degree of steepness (i.e. it is not possible to attend to the "steepest" line).

3) Visual search can be performed as a sequence of guided operations. As discussed above, search for a conjunction of two features can be more efficient than an unguided serial search. For example, in a search for a black vertical item among black horizontal and red vertical items attention can be directed to the intersection of the set of all black items and the set of all vertical items. However, the following task poses a problem:



Now the target can be either white or black and either vertical or horizontal. Guiding attention to the intersection of the set of items that are black or white and the set of items that are vertical or horizontal is obviously useless. Instead, subjects appear to execute two sequential searches, first guiding attention to one target type and then, if no target is found, guiding attention to the other type. This yields RT x set size functions with shallow slopes but higher intercepts. That is, the double searches take somewhat longer but remain efficient.

ATTENTIONAL PRIORITY IN VISION

Steven Yantis
Department of Psychology
Johns Hopkins University

Coherent thought and action concerning visual object configurations require a control mechanism, visual attention, to select task-relevant subparts of the input image for rapid processing. This requirement arises because visual cognition is limited in capacity: there is often more sensory input available than the visual system can handle efficiently at one time. In this paper, I describe some new data that constrain the possible mechanisms for attentional capture, and I propose a model for attentional prioritization that is a hybrid of several recent architectures for visual selection.

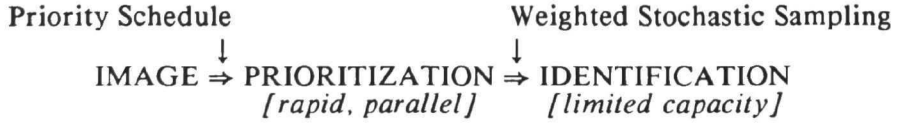
Experiments conducted in our lab reveal that spatiotemporal discontinuities (e.g., abrupt onset or movement) capture attention. In the experiments, subjects search a multielement array for a prespecified target element. When the target is presented with an abrupt onset and nontargets are presented without abrupt onset, then response times are rapid and do not depend on the number of nontargets in the display. When the target is among the objects without abrupt onsets, response times increase monotonically with the number of objects in the display. This suggests that the onset element is processed with high priority (e.g., if decisions are made serially, it is almost always processed first).

Attentional capture by abrupt onset is not absolute, of course. We have found that focussing attention on a spatial location that is likely to contain a target prevents capture by the appearance of an onset elsewhere. This is adaptively sensible, and suggests that there can be multiple criteria for selection, some that are more or less hard-wired and bottom-up (e.g., onsets) and others that are strategic and top-down (e.g., positional expectancies).

Our most recent data concern the mechanism by which attentional priorities are established and maintained when there are several high-priority and several low-priority objects in the image. We have analyzed response times when there are multiple onset and no-onset elements present in a single trial and found that the system queues or tags as many as four onset elements for attentional priority. These elements are processed earlier or receive more computational resources than lower-priority elements do. Preliminary data also suggest that a similar mechanism may operate in other stimulus domains when priority is specified by top-down selection criteria.

A model for attentional priority (see diagram) that can incorporate these results as well as other facts about visual selection incorporates a two-stage architecture, in which the first stage (Prioritization) rapidly and in parallel prioritizes objects in the image based on the extent to which various selection criteria are satisfied (including both bottom-up criteria like onset and top-down criteria like expected position). The importance of any given selection criterion is specified by a priority schedule, and priorities are assumed to be dynamically updated. The second stage (Identification) allocates limited computational resources based on the priorities assigned at the first stage and identifies objects via weighted stochastic sampling. To the extent the selection criteria in combination with the priority schedule effectively guide resources to task-relevant objects, search will be optimal.

A consensus is emerging that two-stage architectures of this type may account for a wide range of visual phenomena. Further empirical work in psychophysics, neuropsychology, and neurophysiology can place additional constraints on the detailed structure of these models.



Schematic diagram of prioritization model.

References

Bergen J. R. and Julesz B. (1983) Rapid discrimination of visual patterns. IEEE Trans on Systems, Man, and Cybernetics, SMC-13 857-863

Cave K. R. and Wolfe J. M. (1990) Modeling the role of parallel processing in visual search. Cognitive Psychology, 22

Duncan J. and Humphreys G. W. (1989) Visual search and stimulus similarity. Psychological Review, 96 433-458

Egeth H. E., Virzi R. A.and Garbart H. (1984) Searching for conjunctively defined targets. J. Exp. Psychol.:Human Perception and Performance, 10 32-39

Klein R. and Farrell M. (1989) Search performance without eye movements. Perception and Psychophysics, 46 476-482

Mackeben M. and Nakayama K. (1988) Fixation release facilitates rapid attentional shifts. Invest. Ophthalmol. Vis. Sci. (suppl), 29

Nakayama K. and Silverman G. H. (1986) Serial and parallel processing of visual feature conjunctions. Nature, 320 264-265

Sagi D. and Julesz B. (1986) Fast noninertial shifts of attention. Spatial Vision, 1 141-149

Treisman A. (1988) Features and objects: the fourteenth Bartlett memorial lecture. The Quarterly J of Experimental Psychology, 40A(2) 201-237

Treisman A. and Gelade G. (1980) A feature-integration theory of attention. Cognitive Psychology, 12 97-136

Treisman A. and Sato S. (1990) Conjunction search revisited. J. Exp. Psychol.:Human Perception and Performance, 16 ??

Treisman A. and Souther J. (1985) Search asymmetry: A diagnostic for preattentive processing of separable features. J. Exp. Psychol. - General, 114 285-310

Tsotsos J. K. (1989) Analyzing vision at the complexity level. Brain and Behavioral Sciences, ms ms

VanEssen D. C. and Maunsell J. H. R. (1983) Hierarchical organization and functional streams in the visual cortex. Trends in Neuroscience, 6 370-375

Wolfe J. M., Cave K. R.and Franzel S. L. (1989) Guided Search: An alternative to the Feature Integration model for visual search. J. Exp. Psychol. - Human Perception and Perf., 15 419-433

Wolfe J. M., Yu K. P., Stewart M. I., Shorter A. D., Friedman S. R.and Cave K. R. (1990) Limitations on the parallel guidance of visual search: Color X Color and Orientation X Orientation conjunctions. J. Exp. Psychol. - Human Perception and Performance, in press

CONCEPTUAL COHERENCE IN TEXT AND DISCOURSE

Arthur C. Graesser
Departments of Psychology and Mathematical Sciences
Memphis State University
Memphis, TN 38152

Description of Symposium

The participants in this symposium present computational and psychological models of coherence in text and discourse. The emphasis is on conceptual coherence rather than syntactic or linguistic coherence. That is, the primary goal is to explain how the text (written or oral), the semantic representation, and world knowledge together determine how idea units are connected coherently. Syntax and other surface linguistic features undoubtedly play a role in a complete theory of coherence; however, their role is not the primary focus of this symposium.

The participants in this symposium span diverse fields in cognitive science: Cognitive psychology, artificial intelligence, computational linguistics, discourse processing, and educational psychology. However, all of the models adopt a computational approach to investigating coherence. That is, each model has particular assumptions about knowledge representation, the formal composition of idea units, symbolic procedures that operate on idea units, and the management of working memory. The models address the process of constructing coherent meaning structures during comprehension in addition to addressing the representation of the meaning structures.

There are several foundations for establishing conceptual coherence when text or discourse is comprehended. First, the textbase becomes more coherent to the extent that explicit propositions can be connected by anaphoric references and bridging inferences (Clark & Haviland, 1977; Halliday & Hasan, 1976; Kintsch & van Dijk, 1978; van Dijk & Kintsch, 1983). Second, a set of text propositions are conceptually related by virtue of higher order packages of generic world knowledge, such as scripts and frames (Bower, Black, & Turner, 1979; Minsky, 1975; Graesser & Clark, 1985; Schank & Abelson, 1977). Third, coherence may be imposed by virtue of abstract rhetorical configurations (Meyer, 1985), such as problem+solution, claim+evidence, assumptions+conclusion, compare+contrast, setting+plot, question+answer, and so on. Fourth, there is a pragmatic level that considers the common ground and goals of the speech participants. All of these foundations are discussed in this symposium.

Symposium participants

Arthur Graesser, symposium chair, Introduction
Richard Alterman, The conceptual coherence of events
Kathleen Dahlgren, Bases of Coherence
Bruce Britton, Kintsch's computational model of coherence processing: Diagnosing inferences in naturally-occurring text, and repairing text accordingly to increase learning
Paul van den Broek, Establishing coherence during reading: A process model of inference generation
Charles Fletcher, A model of narrative comprehension and recall
Roger Kreuz and Richard Roberts, The elements of conversational coherence
Tom Trabasso and Nancy Stein, discussants

References

Bower, G. H., Black, J. B., & Turner, T. J. (1979). Scripts in memory for text. Cognitive Psychology, 11, 177-220.

Clark, H.H., & Haviland, S.E. (1977). Comprehension and the given-new contract. In R.O. Freedle (Ed.), Discourse production and comprehension. Norwood, NJ: Ablex.

Graesser, A. C., & Clark, L. F. (1985). Structures and procedures of implicit knowledge. Norwood, N. J.: Ablex.

Halliday, M.A.K., & Hasan, R. (1976). Cohesion in English. London: Longman.

Kintsch, W., & van Dijk, T. A. (1978). Toward a model of text comprehension and production. Psychological Review, 85, 363-394.

Meyer, B. J. F. (1985). Prose and analysis: Purposes, procedures, and problems. In B. K. Britton, & J. B. Black (Eds.), Understanding expository text. Hillsdale, NJ: Erlbaum.

Minsky, M. (1975). A framework for representing knowledge. In P.H. Winston (Ed.), The psychology of computer vision. New York: McGraw-Hill.

Schank, R., & Abelson, R. (1977). Scripts, plans, goals and understanding. Hillsdale, N. J.: Erlbaum.

van Dijk, T. A., & Kintsch, W. (1983). Strategies of discourse comprehension. New York: Academic Press.

Reasoning about a Semantic Memory Encoding of the Connectivity of Events[1]

Richard Alterman
Computer Science Department
Brandeis University

In artificial intelligence, human understanding of text is modeled by programs which construct representations. A critical question concerns determining the form of the representation (hence the form of the understanding). The notion of coherence emphasizes the connectivity of items in the understanding/representation. Coherence representations either tie together the words of the text, the discourse units of the text, or the concepts the text invokes. The coherence viewpoint is exemplified by the works of Halliday & Hasan (1976) on cohesion, Lockman & Klappholz (1980) on contextual reference resolution, Hobbs (1979) on rhetorical coherence, and the semantic network-derived text representation techniques of Norvig (1989), Kintsch (1988), and Charniak (1986, 1983).

In the theory of *event concept coherence* (ECC: Alterman, 1985;1982), a sequence of events is represented by a copy of a portion of the web of concepts that holds the events together in semantic memory. The basic idea is that semantic memory provides a vocabulary for encoding the events to be understood: the 'language' of the representation is provided by an underlying semantic network. The work on event concept coherence is most directly influenced by early work in semantic networks and spreading activation (Quillian, 1968; Collins & Loftus, 1973), the incorporation of case frames (Fillmore, 1968) into the semantic network framework by Simmons (1973), and the work of Schank (1982) on dynamic memory. I will refer to the ouput derived from a ECC analysis as an event connectivity graph.

The focus of this talk is on the semantics of the ECC-derived representation scheme. I will describe a program called SSS (developed with Larry Bookman) that exploits the event connectivity graph in support of several reasoning tasks: SSS uses the graph to identify the conceptual roots of the narrative. (Roughly the conceptual roots correspond to the basic event notions of the representation/understanding.) An interesting property of the conceptual roots is that they are the minimum set that *covers* the entire interpretation graph. With the addition of the techniques that determine the conceputal roots, SSS is able to succinctly explain the connection between any two concept coherent events in the narrative. Also implemented in SSS is a measure of importance that quantifies the author's conceptual emphasis. Lastly, I will outline the summarization techniques used by SSS to describe the basic event content of the narrative.

[1]This work was supported in part by the Defense Advanced Research Projects Agency, administered by the U.S. Airforce Office of Scientific Research under contract #F49620-88-C-0058.

Bases of Coherence

Kathleen Dahlgren
Intelligent Text Processing, Inc.
1310 Montana Avenue, Santa Monica 90403

Different coherence theories related different entities: portions of the text, propositions expressed by the discourse, or nodes in a semantic net of inferences. We will justify the claim that coherence relations related the discourse events, states and other abstract types in a cognitive model of the events introduced into the discourse. Coherence relations form a cohesive event model for the discourse. "Coherent" means that there is some causal or other link between the content of each clause and some other clause or clauses in the discourse.

Choosing events in a model as the relata assumes a perspective on the nature of the relations themselves. If coherence relations are justified on the basis of cognitive strategies for the interpretation of events, then they must relate events or events in a model. We propose a short closed set of coherence relations based upon cognitive strategies which are employed in the interpretation of events as observed, as well as of events as reported in discourse. They are justified by evidence that they are used in interpretation of observed events, psychological evidence that texts are interpreted using them as the basis for integrating a new sentence interpretation with the prior discourse interpretation, morphological forms, and evidence that they are sufficient to determine segment boundaries which constrain anaphor resolution.

Psychological evidence for coherence includes the intuition that (1) is more acceptable as a discourse than (2).

(1) John invested heavily. He made a huge profit.
(2) John invested heavily. He ate pizza.

Psycholinguistic studies justify coherence in several ways. Subjects reliably draw large numbers of inference about texts when questioned after reading. Those events which are causally or superordinately related to others in a text are more quickly interpreted, better recalled, and better recognized than unrelated events. Texts which are organized for ready identification of event hierarchy (topicality) are more quickly read and better understood. Discourse interpretation involves the construction of an event model of the emerging situation.

Linguistic evidence for discourse coherence is provided by our study of anaphor resolution in a 20,000 word corpus. Pronominal anaphors cannot find their antecedents across segment boundaries. The latter are most often signalled by change in coherence relation and change in sentence subject, not by cue phrases. Therefore, constraints on anaphor resolution provide evidence that coherence relations must be assigned during discourse interpretation.

Kintsch's Computational Model of Coherence Processing: Diagnosing Inferences in Naturally-occurring Text, and Repairing Text Accordingly to Increase Learning

Bruce K. Britton

University of Georgia

Kintsch's computational model of coherence processing provides an algorithm for detecting specific locations in text that require bridging inferences to create coherence. Only when propositions representing those inferences are inserted into the text base can Kintsch's program create a coherent representation of the text.

If Kintsch's model is correct for humans, then texts that require many such bridging inferences should not be represented coherently in memory, and so should be difficult to recall. This prediction was confirmed in a study of eight 600-1000 word history textbook passages. Reliable negative correlations (-.70 to -.89) were found between the number of inferences detected by Kintsch's program and the amount of recall.

Kintsch's model also implies that when such inferences are experimentally inserted into a text that requires them (i.e., the text is repaired according to the principles of the Kintsch model) the new text should be represented more coherently in memory, and therefore should be recalled better; also, the reader should have a better situation model of the situation described by the text, and should read at a faster rate because no pauses are necessary to compute inferences.

In two experiments with undergraduates and Air Force recruits, the Original Version of a 1000 word naturally occurring Air Force text was repaired according to the principles of the Kintsch program. Then the new version, here called the Principled Revision, was empirically compared with the Original Version. Measures of learning and reading rate showed that recall was doubled for the Principled Revision, the situation model was significantly better for the Principled Revision, and reading rate was significantly faster.

These studies support Kintsch's model and its usefulness for diagnosing processing problems in naturally occurring texts, and repairing them to increase learning.

Establishing Coherence during Reading: A Process Model of Inference Generation

Paul van den Broek
Department of Educational Psychology - University of Minnesota
178 Pillsbury Dr. SE - Minneapolis, MN 55455

In this presentation, I will describe a process model of reading comprehension. According to the model the reader attempts to build a coherent mental representation or situation model of the text by maintaining local coherence. Local coherence is maintained via backward and forward causal inferences. Backward inferences connect each focal event (1) to a preceding event that is still in short-term memory, (2) to earlier events that are reinstated from long-term memory, and/or (3) to events that are added to the situation model via elaborative inferences. In addition, forward inferences may anticipate (1) upcoming events, as well as (2) future relevance of the focal event. The model proposes a set of production rules that direct the generation of these inferences. The production rules are based on two types of constraints. Conceptual constraints determine whether sufficient coherence has been established, or whether additional inferences are required before the focal event is adequately comprehended. Thus, the conceptual constraints provide criteria for coherence that initiate and terminate inferential processes. Procedural constraints reflect the processing resources that are available to the reader. Based on attentional or short-term memory limitations, they determine whether long-term memory for the text or general background knowledge is accessed to generate the required inferences. Together, the conceptual and procedural constraints establish under what circumstances causal inferences are generated, what kinds of inferences are made, and what their content is. Reading time and primed recognition data provide empirical support for several aspects of the model.

A Model of Narrative Comprehension and Recall

Charles R. Fletcher
Department of Psychology
University of Minnesota
75 East River Road
Minneapolis, MN 55455

In this presentation, I will describe a process model of narrative comprehension and recall that borrows heavily from existing models of comprehension, problem solving, and retrieval from long-term memory. The comprehension component of the model adds propositions to its short-term memory one at a time. As each proposition is processed, it becomes associated with itself, the context, and all other propositions in short-term memory. When a sentence boundary is reached, the model focuses its attention on the most important propositions from the last clause with causal antecedents--but no consequences--earlier in the text. The free recall component of the model uses a probabilistic search process to retrieve a causal path that connects the text's opening to its final outcome. This is achieved by using propositions from the most recently recalled clause with antecedents--but no consequences--as retrieval cues. A computer simulation of this model accounts for 35% of the variance in college students' free recall data on four simple narrative texts.

The Elements of Conversational Coherence

Roger J. Kreuz and Richard M. Roberts
Memphis State University

Researchers who study coherence have primarily focused upon text in their efforts to determine how coherence is established and maintained. Coherence processes in conversation are, to a certain degree, dependent upon these same mechanisms. However, conversations differ from texts in important ways, and these discrepancies have important implications for our understanding of conversational coherence. Some of the ways in which texts and conversations differ will be reviewed.

In addition, the concept of conversational coherence itself will be critically examined. A lack of consistency in the literature has made it difficult to operationally define coherence. However, a simple metric for determining the loss of conversational coherence will be advanced and explained.

Two fundamentally different ways of viewing conversation will be presented. One view suggests that conversational coherence is fragile, while the other maintains that coherence is relatively robust. Some evidence exists for both of these extremes, and this discrepancy will be resolved by positing that coherence is best viewed as a relative attribute of conversation, instead of as an absolute attribute.

Previous research on conversational coherence will also be reviewed. In particular, Paul Grice's ideas about a Cooperative Principle and conversational maxims will be characterized as an important but incomplete description of conversational interaction. By appealing to Clark's notion of common ground (in which the degree of experiential overlap is computed by conversational participants), a number of problems concerning conversational coherence be resolved.

A New Look at Decision Making

Symposium Organizers:

Susan Chipman, Office of Naval Research

Judith Orasanu, Army Research Institute and Princeton University

Academic research on decision making appears to have become boxed into a rather narrow and restrictive paradigm. Recently some decision researchers introduced their work approximately as follows: "A decision problem consists of three basic elements: (1) the alternatives available to the decision maker; (2) probabilities of events that relate outcomes to the choice of alternatives; and (3) the values associated with the outcomes." Much research has addressed the problems that arise when human beings are asked to carry out a rational decision analysis within that framework, the difficulties that people have in estimating probabilities and values at all, let alone consistently.

Perhaps those difficulties should cause us to ask some more fundamental questions. In life, how often do decision situations resemble "decision problems" as formulated in the psychological laboratory? Although every action could be said to entail a decision, it seem rare that life lays out before us a clearly defined problem accompanied by a finite set of alternatives, and then pauses long enough to permit a full rational analysis. Rather, perceiving that there is a problem which calls for a decision and thinking of any reasonable alternative that might be taken seems more the issue. Is it even true that decisions pass through a representation that resembles the laboratory problem? No clear evidence exists that they do. Of course, traditional decision researchers can retreat to the position that this is the way decision making ought to occur, even if in fact it does not.

When life does present clear occasions for decisions, and sometimes even moderately clear alternatives, time is often very limited. Views of the way decisions ought to be made often fail to take that factor into account. Seven minutes in July. That is the time that elapsed in the entire Vincennes incident, from the first report of a possible contact with a plane until the airliner was shot down. Is it reasonable to think about a decision-theoretic analysis in that time period? Even if it were, on what basis would one assign probabilities to the possible identities of the aircraft? What is the proper cost to assign to the possible damage to one's own ship and the possible deaths of one's own men? What is the proper cost to assign to the possible deaths of the passengers on an aircraft? As is so often true, these values and probabilities seem very difficult to assign. Whatever the rational appeal of the process, the outcome remains uncertain. Mistakes -- as seen in hindsight -- will be made.

Yet, as we shall hear in this symposium, many important decisions must be made in less than seven minutes -- decisions about what to do for a patient arriving in an emergency room, decisions about how to combat a rapidly developing fire. Most such decisions are made in less than one minute.

In this symposium, we present several samples of newer approaches to decision making, approaches that go beyond the bounds of the neat and clean laboratory "decision problem." We shall see that understanding decision making outside the laboratory is a challenge that calls for all the methods and techniques that the full range of cognitive science has to offer. This undertaking is necessary in order to develop valid and useful methods for training people to make more effective decisions and to design effective and useful aiding systems.

The first presentation is Gary Klein's naturalistic studies of decision making processes as they actually occur in time-pressured life-and-death situations, the source of the data cited above. These studies are leading to the conclusion that decision processes of expert performers in these situations involve perceptual recognition of the type of situation they face, which triggers appropriate actions suggested by past experience, which are then evaluated in rapid imaginal processes and patched if necessary.

If decision making involves perception, psychology has a rich repertoire of psychophysical techniques for figuring out what is going on in such processes. The presentation by John Swets displays an elegant orchestration of those research techniques in order to understand and aid the process of classifying mammograms.

But some situations are more complex, or at least very different in kind, from those that can be understood in terms of a set of perceptual or conceptual patterns. Sometimes the decision maker needs to make sense of a diverse collection of facts, to mold them into something like a consistent story or scientific theory. Paul Thagard will discuss the extension of his approach to the self-consistency of theories to understanding what occurs in decision-making situations.

For several years, Marvin Cohen has worked on the problem of creating practical, usable, and acceptable decision aids for time- pressured military decision making. He will discuss how it has been possible to make use of past decision making research and where the serious gaps lie, where the existing body of research knowledge has little to say.

Finally, it is true that many, if not most, critical decision making situations involve several actors. Judith Orasanu will describe on-going research into the decision processes and problems revealed in the tape recordings of flight crews faced with critical problems.

Participants:

Susan Chipman, ONR, Chair
Gary Klein, Klein Associates
John Swets, Bolt Beranek & Newman, Inc.
Paul Thagard, Princeton University
Marvin Cohen, Decision Sciences Corporation
Judith Orasanu, ARI and Princeton University

Recognitional Decision Making in Natural Situations

Gary Klein

Klein Associates

A model of proficient decision making will be described that emphasizes situation assessment rather than a comparative evaluation of different options. The model is a descriptive account of research on skilled decision makers in different domains. Research will be presented that suggests that proficient decision makers rarely contrast options; rather, experience enables decision makers to generate a plausible course of action as the first they consider. In contrast, the novices we studied were more apt to generate a set of options for careful evaluation. Research will be reviewed from areas including urban firefighting, forest fire operations, tank platoon maneuvers, battle planning, and training device design. These are domains marked by high time pressure, ill-defined goals, ambiguous and incomplete information, high stakes, and personal responsibility for outcomes. In all, we have examined over 450 decision points that were examples of nonroutine and difficult incidents, and we found that a recognitional strategy generalized across domains. One of the important aspects of recognitional decision making is the use of mental simulation to evaluate options without having to contrast strengths and weaknesses. The presentation will address the role of the simulation heuristic within a decision making framework.

Improving Perception-Based Decisions

John A. Swets

BBN Laboratories
Cambridge, Massachusetts

Images and other visual representations are used to make diagnostic decisions about underlying conditions in many fields, including clinical medicine, materials testing, geological prospecting, and national defense. In many cases, it may prove possible to describe, refine, and improve the decision processes by viewing them as consisting of numerical assessments of the relevant perceptual features of an image and a merging of the feature assessments with appropriate weights to yield an overal diagnostic probability. I describe how certain psychological and statistical techniques can be used to develop decision aids for such diagnostic tasks, specifically in the context of mammography for the diagnosis of breast cancer.

Specialists in mammography participated in a four-step procedure: 1. individual interviews produced a comprehensive list of features; 2. similarity judgments of pairs of representative cases were subjected to multidimensional scaling to refine the feature list; 3. a consensus meeting refined feature names, descriptions, and rating scales; 4. ratings of a set of known positive and negative cases were submitted to discriminant analysis to produce a minimal but sufficient set of effective features and their optimal weights. The resulting decision aids were a checklist of features with rating scales and a computer-based (discriminant analysis) algorithm that accepted the ratings for a case and issued an estimate of the probability of malignancy. In a test of the aids, general radiologists read a set of known cases first in their standard manner and then with the aids. The aids produced a substantial improvement in accuracy.

The presentation will conclude by opening a discussion of the extent to which this approach might generalize to situations with conceptual as well as perceptual features, and with a discussion of possible limits on feature representations.

Explanatory Coherence and Naturalistic Decision Making

Paul Thagard

Cognitive Science Laboratory, Princeton University

This talk will describe the relevance of my theory of explanatory coherence (*Behavioral and Brain Sciences*, 1989) for decision making in cases where decisions depend on evaluation of competing hypotheses. In complex cases it is often necessary to form and evaluate hypotheses concerning the nature of the situation. For example, a fire chief may need to infer the source and nature of a fire before deciding how best to fight it. Judges and juries are frequently called upon to evaluate explanatory hypotheses in criminal trials, asking, for example, whether the proposition that the accused murdered the deceased is the best explanation of the death and other evidence. But inference to the best explanation in such cases is not just a matter of considering what hypothesis explains the most evidence, since it is standard in trials to consider a *motive* that could explain why the murder was committed. The acceptability of a hypothesis increases on the basis of there being explanations of it, as well as on the basis of what it explains. Everyday decisions that involve other people often require explanatory inferences concerning their beliefs, desires, and intentions. In adversarial situations such as competitive games, business, diplomacy, and war, it is often necessary to infer the plans of the adversary. Plans can sometimes be inferred as part of the best explanation of what the adversary has done so far.

The theory of explanatory coherence applies naturally to cases such as these. The theory is implemented in ECHO, a program that takes input about explanatory relations to create networks of hypotheses. It then performs parallel constraint satisfaction to evaluate hypotheses using standard connectionist algorithms. ECHO has been used to analyze the decision made in July 1988 by Captain Rogers of the USS Vincennes to shoot down what appeared to be an attacking aircraft. The true hypothesis that the plane was a commercial airliner was considered and rejected in favor of the hypothesis that the plane was an attacking F-14.

Cognitive Strategies and Adaptive Aiding Principles in Submarine Command Decision Making

Marvin S. Cohen

Decision Science Consortium, Inc.

Decision aiding efforts have often been premised on the assumption that unaided decision making is subject to fundamental flaws or "biases," which can be corrected only by adoption of "normative" methods such as Bayesian decision analysis. Such an approach may force decision makers to adopt highly unfamiliar modes of reasoning; as a result, aids may not be used, or if used, may fail to exploit user knowledge. An alternate approach is to start with the user's preferred way of solving the problem and to examine carefully its strengths and weaknesses. Aids are then designed which support more optimal *iants* of the user-preferred strategy. This approach, called Personalized and Prescriptive Aiding, has recently been applied to submarine tactical decision making. Experiments were conducted to investigate the decision strategies adopted by submarine staff in handling multiple goals (attacking high-value targets versus avoiding counter-detection), combining multiple uncertain estimates (of target range), and dealing with ambiguous probability assessments (of hit and counterdetection). In each case, a significant number of subjects failed to aggregate information into a single abstract measure as required by the standard normative approaches to these problems (i.e., a measure of expected utility, a pooled target range, or an expected probability). Neither the results nor the comments of subjects were consistent with the "psychophysical" approach to decision biases proposed by Prospect Theory (Kahneman and Tversky, 1979), with a workload-versus-accuracy tradeoff hypothesis (Johnson and Payne, 1984), or with individual-differences models (Lopes, 1987). The findings are best understood as reflecting the adoption by subjects of simplifying assumptions in order to exploit their knowledge of the problem environment. Decision makers appear to utilize a basic or natural level of representation (Rosch, 1976)--in terms of specific evaluative dimensions, range estimates, or conditional probabilities--that effectively captures their causal or correlational knowledge. Decision aiding concepts have been developed and demonstrated for range pooling and attack planning, which permit users to adopt such assumptions while guarding against potential pitfalls in their preferred decision-making strategy; rather than demanding adoption of an abstract numerical representation, such aids keep track of assumptions, amke users aware of alternatives, and actively warn them when alternatives have significantly different implications.

Shared Mental Models and Crew Decision Making

Judith Orasanu

U.S. Army Research Institute and Princeton University

When faced with decisions in high-stress situations, groups, like individuals, typically narrow the range of information they consider and revert to dominant behavior patterns. Yet some groups are clearly more successful than others in coping with emergencies. Compared to individual decision makers, groups offer expanded cognitive resources, but these contribute to increased effectiveness only if they are appropriately orchestrated and exploited. The research described here aims to account for differences in group decision making effectiveness in critical situations using process tracing and discourse analysis techniques.

Subjects were experienced airline cockpit crews making decisions about how to handle realistic emergencies, such as hydraulic failure or low fuel in bad weather, in high fidelity simulators. The hypothesis addressed in this work is that effective crews develop shared mental models and use language to support joint cognitive functions critical to decision making. These include situation assessment, interpreting information, constraining solutions, anticipating future events, and justifying and priming future actions. We hypothesize that the net result of developing shared models is to reduce information processing demands during high workload periods, permitting better decisions and assuring coordinated actions. The presentation will conclude with implications for training and aiding groups for decision making.

The study of expertise: Prospects and limits

A Symposium organized by Anders Ericsson and Vimla Patel

During the last twenty years the study of expertise has emerged as one of the major research topics in Cognitive Science. Innovative and diverse research approaches have accumulated a rich body of differences between experts and novices in many different domains of expertise, such as chess, bridge, physics, medicine, writing, music, decision making, computer programming and sports. The majority of the findings suggest similar characteristics of expertise across domains; but some findings, for example, imply that expertise is not always associated with superior performance (cf. expert decision makers) or with superior memory performance (cf. some types of sport, music and medicine). The purpose of this symposium is to provide a state-of-the-art review of contemporary research on expertise to identify methodological issues, theoretical mechanisms and empirical findings generalizing across different domains of expertise.

The paper by Ericsson discusses different research approaches to the study of expertise and adopts the original expertise approach derived from de Groot's and Chase and Simon's pioneering research on chess expertise as the most fruitful framework for the study of expertise. According to this approach, superior real-life performance of experts has first to be captured by specially designed tasks in the laboratory. The superior performance of experts, which can be reliably reproduced in the laboratory, can then be analyzed in terms of superior mediating processes using process tracing and experimental analysis. The focus in this approach on stable expert performance may be viewed as both a strength as well as a limitation of the original expertise approach.

The research findings from analyses of expert performance in chess (Charness), physics (Anzai), medicine (Patel & Groen) and sports (Allard & Starkes) are reviewed and discussed in terms of generalizable concepts and phenomena across the four domains of expertise. Charness focuses on the balance between knowledge and search in selection of the correct actions (e.g., chess moves). Anzai describes the representation of physics problems as captured by the construction of diagrams. Patel and Groen discuss forward reasoning and enhanced recall in medical expertise. Allard and Starkes examine the relation between superior perceptual and memory performance and level of expertise for a wide range of different types of sports. The symposium is concluded with a general discussion by Keith Holyoak.

Presenters: Anders Ericsson
Neil Charness
Yucho Anzai
Vimla Patel
Fran Allard
Discussant: Keith Holyoak

Approaches to the empirical study of expertise: Some general issues and considerations

K. Anders Ericsson
University of Colorado at Boulder

With the growing interest in research on expertise, a diverse set of research approaches has emerged. Experts are often identified using social criteria and the amount of experience and the level of performance of these experts is then compared to those of novices and less experienced subjects on various tasks. In this paper, we argue that the success of the pioneering research on chess was due to the satisfaction of a number of additional constraints. The primary focus of the pioneering research on expertise in chess was to study stable superior performance in real life expert activities. Hence the first step in such a study involves an analysis of that real life performance to identify a set of tasks which captures reliably the superior performance of experts in the laboratory. Once the superior performance of experts has been reliably reproduced in the laboratory, examination of the critical mediating processes responsible for the superior performance can be made using standard methods of process tracing and experimental analysis. When the critical processes have been identified, the issue of how these processes can be acquired during extended practice is raised. Theoretical analyses of learning and skill acquisition are explored. Most importantly, empirical studies of the acquisition of the critical processes can be made by cross-sectional comparisons of subjects with differing amounts of expertise and experience, by studies of extended training on the target tasks and by a careful analysis of the real-life practice activities designed to maximize improvement of performance.

Skill in Chess: The Balance Between Knowledge and Search

Neil Charness
University of Waterloo

Two strikingly different routes have been taken to achieve high levels of chess skill. Programs such as Deep Thought use extensive search (about 720,000 positions examined/second) to decide on the best move, with little knowledge of chess beyond center control, mobility, and piece balance. Human grandmasters rely on extensive knowledge of chess openings, middlegame plans, and techniques for playing endings to enable extremely modest search (10 positions/minute) to uncover the best move. I will examine the trade-offs between knowledge and search in chess and focus on the knowledge base that humans rely on. Questions to be addressed are: 1) How much knowledge accrues to humans via tournament play versus study? 2) What effect have national training programs had on the development of skilled players? 3) How do the different forms of chess (blitz, tournament, postal) depend on search versus knowledge? 4) How has knowledge accumulation affected peak levels of chess performance by Grandmasters over the past century? 5) What role does aging play in the inverted U-shaped life-time chess performance function?

Learning How to Draw Diagrams for Physics Expertise

Yuichiro Anzai
Keio University

Experts in physics draw diagrams that include appropriate information for solving problems. While the diagrams drawn by novice learners are usually not well suited for problem solving. Why does this difference in expertise occur? How does a novice acquire the knowledge of how to draw diagrams? I try to answer these questions by providing a cognitive theory that explains the process of learning to draw physics diagrams. In short, the theory insists that the two kinds of knowledge, for how to draw diagrams and how to make inferences in problem solving, are acquired through the process of utilizing each other. [That is, a more sophisticated inference strategy for problem solving, once acquired, aids the acquisition of a more elegant procedure for representing problems with diagrams and vice versa.] Thus, the two processes, learning inferencing and drawing, may bootstrap to make expertise grow. The theory was developed based on observations from a longitudinal study of one novice subject acquiring expertise.

The generality of medical expertise: A critical look

Vimla L. Patel and Guy J. Groen
McGill University

Two fundamental empirical findings in research on expert-novice comparisons have been the phenomena of enhanced recall and of forward reasoning. The first refers to the fact that experts have superior skills in recognizing patterns in their domain of expertise. The second pertains to the findings that, in solving routine problems in their domains, expert problem solvers tend to work forward from the given information to the unknown. Our main purpose is to re-evaluate, in the context of medical expertise, the assumption that there is a close relationship between these two phenomena, in the context of expert-novice comparisons of clinical reasoning in medicine.

We present evidence to show that recall of clinical information by medical specialists is unrelated to pure diagnostic accuracy. In contrast, pure forward reasoning is closely related to diagnostic accuracy. It is shown that this pattern is a function of the specific relevant knowledge possessed by the physicians. It is also shown that it is dependent on the structure of the clinical case since the presence of "loose-ends" unrelated to the main diagnosis is frequently accompanied by backward reasoning.

The absence of recall differences in experts is contrasted with the recall of intermediates and novices, where strong recall differences are found. This leads to a suggestion that there are two kinds of expertise; generic expertise, in which ways of

representing medical knowledge are acquired, and specific expertise, in which no fundamental representational changes are taking place. The possibility that the acquisition of generic expertise is non-monotonic is also discussed.

Expertise in Sport

Fran Allard, University of Waterloo
and
Jan Starkes, McMaster University

Success in sport involves expertise in both knowledge of the domain and performance of relevant motor skills. For closed motor skills such as dance or figure skating, the skilled individual must be able to remember long sequences of movement elements to perform a role or routine. For open skills such as basketball and football, the ability to learn offensive and defensive patterns is essential for all team members.

Studies will be reviewed to illustrate the superior recall performance shown by expert ballet dancers and figure skaters over less skilled performers when recalling choreographed dance or skating sequences. As in many other skill domains, the experts' advantage evaporates when the sequences presented for recall are randomly ordered strings of the same set of items.

Skilled basketball players show a similar recall superiority for the recall of schematic diagrams of basketball plays, showing that sensitivity to the structure of the domain cuts across both open and closed sport skills. One exception to the finding of the importance of pattern for sport experts occurs for volleyball. Skilled volleyball players do not show differential recall accuracy for either schematic play recall or for recall of the position of players on slides of real volleyball games. Expert volleyball players are much faster than less skilled players at detecting the presence of a ball in briefly presented slides of volleyball games, showing rapid search to be an important component to skill in this game.

The relationship between cognitive skill and performance skill will also be considered in this presentation.

Author Index

Adelson, Beth 9, 253
Agre, Phil 694
Aha, David W. 141
Ahmad, Subutai 629
Ahn, Woo-Kyoung 149
Ajjanagadde, Venkat 285
Alterman, Richard 908
Anderson, John R. 702, 796
Arbib, Michael A. 598
Arens, Yigal 487
Avrahami, Judith 125

Baars, Bernard 828
Badler, Norman I. 412
Ball, William 717
Bapi, Raju S. 915
Bateman, John A. 471
Bates, Elizabeth A. 404
Bauer, Malcolm I. 317
Biederman, Irving 614
Birnbaum, Lawrence 245, 954
Boyle, C. Franklin 788
Braisby, Nick 438
Brand, Matthew 245
Brent, Michael R. 428
Britton, Brent C.J. 637
Bruder, Gail A. 947
Burns, Bruce D. 45

Cardie, Claire 511
Chaffin, Roger 420
Chalmers, David J. 340
Chandrasekaran, B. 670
Chee, Yam San 1
Cleeremans, Axel 709
Clement, Catherine A. 962
Cohen, Jonathan D. 820
Collins, Gregg 954
Corbett, Albert T. 796
Crawley, Rosalind A. 534
Cushing, Steven 812

DeLoache, Judith 22
Dent, Lisa 780
Domeshek, Eric A. 844
Druhan, Barry B. 717
Dunbar, Kevin 93, 333
Dykstra, Dewey I. 788

Elvgren, Gillette 348

Fain Lehman, Jill 804
Falkenhainer, Brian 69
Fay, Anne L. 333
Fehling, Michael 828
Feldman, Jerome A. 686
Fellbaum, Christiane 420
Fisher, Charles 828
Forbus, Kenneth D. 61
Foster, Dean 133
Franks, Bradley 438
Furnas, George W. 662

Gentner, Dedre 22, 38
Gibson, Edward 372
Girotto, Vittorio 293
Goebel, Rainer 852
Goldstone, Robert L. 141
Gonsalves, Renison J. 463
Gray, Kenneth C. 30
Greeno, James G. 77, 939

Hare, Mary 364
Harris, Catherine L. 404
Hemforth, Barbara 558
Hendler, James 550
Hinrichs, Thomas R. 900
Hofstadter, Douglas R. 174
Holyoak, Keith J. 190
Hollbach Weber, Susan 686
Hovy, Eduard 487
Hummel, John E. 614
Hunter, Lawrence 261

Jackson, Douglas 939
Jenison, Rick L. 860
Jennings, Peggy L. 876
Johnson, Hollyn M. 757
Jones, Eric K. 269
Jones, Gregory V. 750
Jones, Jacqueline A. 892, 923

Kalita, Jugal K. 412
Kanwisher, Nancy G. 606
Kareev, Yaakov 125
Keele, Steven W. 876
Kennedy Hickman, Angela 53
Kim, Hyun-Kyung 117
Klahr, David 333
Kleinman, David 534
Kokinov, Boicho Nikolov 741
Kolen, John F. 868

Kolodner, Janet L. 978
Korpi, Margaret 939
Kortge, Chris A. 764
Kowalski, Bernadette 717
Krulwich, Bruce 954

Laird, John 582
Lakoff, George 686
Larkin, Jill H. 53
Leake, David B. 325
Legendre, Géraldine 388, 884
Legrenzi, Paolo 293
Lehnert, Wendy 511
Levine, Daniel 915
Lytinen, Steven L. 503

MacAndrew, Siobhan B.G. 750
Markman, Arthur B. 38
Martin, Joel D. 166
Mastaglio, Thomas 301
Mathews, Robert C. 733
McCartney, Robert 101
McClelland, James L. 709
Michalchik, Vera 939
Miikkulainen, Risto 447
Miller, Benjamin O. 892, 923
Mitchell, Donald H. 215
Mitchell, Melanie 174
Miyata, Yoshiro 388, 884
Monarch, Ira A. 788
Morgan, Stephen T. 590
Mozer, Michael C. 356
Murray, Kenneth S. 309

Narayanan, N. H. 670
Nersessian, Nancy J. 77
Nicholl, Sheldon 455

Oblinger, Dan 61
Oden, Gregg C. 860
Ohlsson, Stellan 237
Omohundro, Stephen 629
Owens, Christopher 277

Patel, Vimla 931
Pazzani, Michael J. 221
Penberthy, Theresa Louise 978
Peters, Sandra L. 157
Pollack, Jordan B. 868
Pratt, Ian 654

Qin, Yulin 646
Quilici, Alex 519
Quinn, Clark N. 16

Ram, Ashwin 198, 206
Rapaport, William J. 157
Rattermann, Mary Jo 22
Redmond, Michael 970
Reeves, Brent 301
Reiser, Brian J. 317
Riloff, Ellen 511
Robertson, Scott P. 542
Roussel, Lewis G. 733

Sandon, Peter A. 574
Scarborough, Don L. 892, 923
Schlimmer, Jeffrey C. 780
Schneider, Walter 566
Schooler, Lael J. 702
Schunn, Christian D. 93
Segal, Erwin M. 526
Seifert, Colleen M. 30, 757
Servan-Schreiber, David 820
Servan-Schreiber, Emile 182
Shedden, Judith M. 566
Shrager , Jeff 694
Shultz, Thomas R. 229
Silverstein, Glenn 221
Simon, Herbert A. 646
Sivaramakrishnan, Malathi 931
Skalak, David B. 109
Slade, Stephen 836
Small, Steven L. 479
Smolensky, Paul 388, 884
Spohrer, James C. 725
Stevenson, Rosemary J. 534
Stevenson, Suzanne 396
Stolcke, Andreas 686
Strube, Gerhard 558
Sycara, Katia P. 85

Tanaka, James W. 622
Theios, John 590
Touretzky, David S. 348

VanLehn, Kurt 717

Waldmann, Michael R. 190
Wang, DeLiang 598
Wanner, Leo 471
Ward, Nigel 380
Weber, Kimberly C. 542
Wheeler, Deirdre W. 348
Wiebe, Janyce M. 947
Wiesmeyer, Mark 582
Wilkins, David C. 455
Wilson, Garry 534
Wrobel, Heike 558

Wu, Horng Jyh P. 503

Yanikoglu, Berrin A. 574
Yaniv, Ilan 133

Zeitz, Colleen M. 772
Zito-Wolf, Roland 908
Zukerman, Ingrid 495

Symposia Index

Abney, Steven P. 1021
Agre, Phil 1006
Allard, Fran 1070
Alterman, Richard 1007, 1054
Anzai, Yuichiro 1069

Baillargeon, Renee 1015
Berwick, Robert C. 1020
Billman, Dorrit 989
Bizzi, Emilio 986
Bobick, Aaron 1018
Britton, Bruce K. 1056
Bullock, Daniel 1030

Caplan, David 1043
Carbonell, Jaime G. 997
Carpenter, Gail A. 1032
Cavanagh, Patrick 1046
Charness, Neil 1068
Chipman, Susan 1060
Cohen, Marvin S. 1065
Crain, Stephen 1044

Dahlgren, Kathleen 1055
Diamond, Adele 1016
Dorr, Bonnie J. 1022

Ericsson, Anders 1067, 1068

Fain Lehman, Jill 1035
Firby, R. James 1008
Fisher, Douglas 989
Fletcher, Charles R. 1058
Fong, Sandiway 1023

Gazzaniga, Michael S. 1043
Gil, Yolanda 997
Gluck, Mark 989
Glymour, Clark 1026
Gopnik, Myrna 1044
Grasser, Arthur C. 1052
Groen, Guy J. 1069
Grossberg, Stephen 1027, 1028, 1030, 1032
Gwiazda, Jane 1014

Hammond, Kristian J. 1005, 1009
Held, Richard 1013
Higginbotham, James T. 988
Hollander, Michelle 1044
Huffman, Scott B. 1035

Jepson, Allan 1019
John, Bonnie E. 1035
Johnson, Mark 1024
Jordan, Michael I. 987
Joseph, Robert 997

Kelly, Kevin T. 1026
Kim, John 1044
Klein, Gary 1062
Knoblock, Craig A. 997
Kosslyn, Stephen M. 1043
Kreuz, Roger J. 1059

Laird, John E. 1035
Langley, Pat 989
Lasnik, Howard 988
Lewis, Richard L. 1035

Marcus, Gary 1044
McCarthy, John 988
Mingolla, Ennio 1028
Minton, Steve 997

Nakayama, Ken 1019, 1047
Newell, Allen 1035
Orasanu, Judith 1060, 1066

Patel, Vimla L. 1067, 1069
Pazzani, Michael 989
Pesetsky, David 988
Pinker, Steven 1044
Posner, Michael I. 1043
Prasada, Sandeep 1044
Prince, Alan 1044

Richards, Whitman 1017
Roberts, Richard M. 1059
Rosenbloom, Paul S. 1035
Rudd, Michael 1028

Simmons, Reid G. 1008
Simon, Tony 1035
Squire, Larry 1044
Stabler, Edward P., Jr. 1025
Starkes, Jan 1070
Swets, John A. 1063

Tessler, Shirley G. 1035
Thagard, Paul 1064
Treisman, Ann 1046

Ullman, Michael 1045

van den Broek, Paul 1057
Veloso, Manuela M. 997

Wexler, Kenneth 1044
Wolfe, Jeremy M. 1048

Yantis, Steven 1050

Visit our Booth in Boston

Instant Access to Information

Human Motor Control

David A. Rosenbaum

October 1990
ISBN: 0-12-597300-4

Cognitive Neuropsychology

A Clinical Introduction

Rosaleen A. McCarthy and Elizabeth K. Warrington

In Paperback: $29.95 (T)
ISBN: 0-12-481846-3
Casebound: $65.00 (T)
ISBN: 0-12-481845-5
September 1990, c. 416 pages
(T) indicates a tentative price.

Neural Models of Plasticity

Experimental and Theoretical Approaches

edited by
John H. Byrne and William O. Berry

In Paperback: $29.95
ISBN: 0-12-148956-6
Casebound: $70.00
ISBN: 0-12-148955-8
1989, 438 pages

Temporal Representation and Inference

Barry Richards, Inge Bethke, Jon Oberlander, and Jaap van der Does

A Volume in the COGNITIVE SCIENCE Series

1989, 365 pages, $39.95
ISBN: 0-12-587770-6

Observer Mechanics

A Formal Theory of Perception

Bruce M. Bennett, Donald D. Hoffman, and Chetan Prakash

1989, 297 pages, $29.95
ISBN: 0-12-088635-9

Case-Based Planning

Viewing Planning as a Memory Task

Kristian J. Hammond

A Volume in the PERSPECTIVES IN ARTIFICIAL INTELLIGENCE Series

1989, 277 pages, $34.95
ISBN: 0-12-322060-2

Cognitive Engineering in Complex Dynamic Worlds

From a Special Issue of the International Journal of Man-Machine Studies

edited by
Erik Hollnagel, Guiseppe Mancini, and David D. Woods

1989, 308 pages, $39.95
ISBN: 0-12-352655-8

Working Models of Human Perception

edited by
Ben Elsendoorn and Herman Bouma

1989, 532 pages, $55.00
ISBN: 0-12-238050-9

Thinking

Directed, Undirected and Creative

K.J. Gilhooly

In Paperback: $24.95
ISBN: 0-12-283484-4
Casebound: $49.95
ISBN: 0-12-283483-6
1988, 236 pages

Computational Models of Learning in Simple Neural Systems

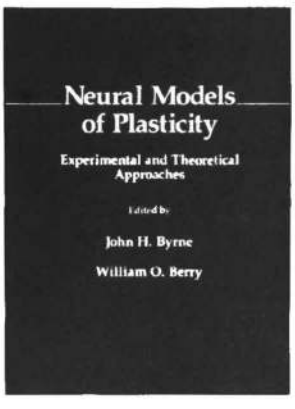
edited by
Robert D. Hawkins and Gordon H. Bower
Volume 23 of PSYCHOLOGY OF LEARNING AND MOTIVATION

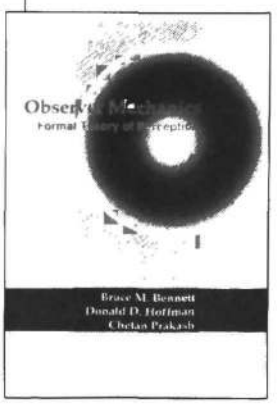
In Paperback: $29.50
ISBN: 0-12-531958-4
Casebound: $59.50
ISBN: 0-12-543323-9
1989, 321 pages

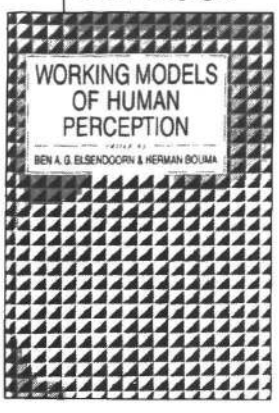
Cognitive Psychology of Planning

edited by
J.-M. Hoc
A Volume in the COMPUTERS AND PEOPLE Series

1988, 252 pages, $39.95
ISBN: 0-12-350770-7







Order from your local bookseller or directly from

ACADEMIC PRESS
Harcourt Brace Jovanovich, Publishers
Book Marketing Dept. #03080
1250 Sixth Avenue
San Diego, CA 92101

CALL TOLL FREE
1-800-321-5068

FAX: 1-314-528-5001
Quote this reference number for *free postage and handling* on your prepaid order ➨ **03080**

Prices subject to change without notice. ©1990 by Academic Press, Inc. All Rights Reserved.
LK/MJD — 03080

New from MIT/Bradford Books

An Invitation to Cognitive Science

edited by Daniel N. Osherson

This major three-volume work provides an integrated treatment and general introduction to the issues, theories, and data of contemporary cognitive science. Language competence and use, visual processing, motoric skill, and thinking are all covered - from neural mechanisms to behavior. Developmental issues and philosophical issues are also included.
3 volume set, 937 pp., illus.
$55.00 paper

Language

An Invitation to Cognitive Science, Volume 1
edited by Daniel N. Osherson and Howard Lasnik
273 pp. $37.50 cloth, $18.95 paper

Visual Cognition and Action

An Invitation to Cognitive Science, Volume 2
edited by Daniel N. Osherson, Stephen M. Kosslyn and John Hollerbach
356 pp. $37.50 cloth, $18.95 paper

Thinking

An Invitation to Cognitive Science, Volume 3
edited by Daniel N. Osherson and Edward E. Smith
308 pp. $37.50 cloth, $18.95 paper

Naturally Intelligent Systems

Maureen Caudill and Charles Butler

"Just what you need to know about neural networks!" **— Bernard Widrow, President, International Neural Network Society**
304 pp., illus. $19.95

Foundations of Cognitive Science

edited by Michael I. Posner

"This volume should be on the bookshelf of every researcher in cognitive science. It is a road map and a guidebook, a dictionary and a pocket encyclopedia. The all-star group of authors adds invaluable insights and perspective to their careful reviews of many literatures." **— S. M. Kosslyn, Harvard University**
888 pp., 87 illus. $45.00

Original in Paperback

The Cerebral Cortex of the Rat

edited by Bryan Kolb and Richard C. Tees

"This volume fills a great need for authoritative information on the structural and functional organization of the rodent cerebral cortex. The contributors are all leaders in the field, and I believe that this compendium of their collective wisdom will become an essential text for anyone who uses rodents as a model of cortical function and disease." **— Patricia Goldman-Rakic, Yale University School of Medicine**
640 pp., 134 illus. $65.00 cloth, $35.00 paper

Visual Agnosia

Disorders of Object Recognition and What They Tell Us about Normal Vision
Martha J. Farah

"Farah presents a lucid and scholarly overview of the visual agnosias. . She examines the problem from the vantage point of a cognitive psychologist and manages to provide a clear distillate of what has historically been a complex and confusing subject." **— V.S. Ramachandran, M.D., University of California at San Diego**

"Every page is rewarding." **— Anne Treisman, Department of Psychology, University of California, Berkeley**
200 pp., illus. $25.00

Connectionist Modeling and Brain Function

The Developing Interface
edited by Stephen José Hanson and Carl R. Olson

"This volume makes a convincing case that data-rich brain scientists and model-rich cognitive psychologists can and should talk to one another. The topics they discuss together here – memory and perception – are of vital interest to both, and their collaboration promises continued excitement along this new scientific frontier." **— George Miller, Princeton University**
423 pp., 122 illus. $37.50

Neural Network Design and the Complexity of Learning

J. Stephen Judd

"Judd's research addresses the computational complexity of learning in artificial neural networks. He has obtained novel theoretical results that provide much-needed guidance for the enterprise of exploring learning within this framework." **— Andrew G. Barto, University of Massachusetts, Amherst**
150 pp., illus. $24.95

The Organization of Learning

Charles R. Gallistel

"Gallistel analyzes fundamental concepts of animal cognition. (and) proposes new and imaginative hypotheses about brain and mental processes and provides new insights about animal behavior." **— Russell M. Church, Brown University**
648 pp., 50 illus. $45.00 (August)

Computational Neuroscience

edited by Eric L. Schwartz

Thirty original contributions review this emerging field in historical and philosophical overviews and in stimulating summaries of recent results. 441 pp. $45.00

Becoming a Brother

Morton J. Mendelson

Mendelson presents a useful general model to account for the early interactions and emerging relationship between a child and a new brother or sister. The model focuses on the child's self-concept, interactive skills, and motivation to interact with the new sibling and reveals the ways in which the acquisition of the sibling role may be influenced by family members and by family relationships.

300 pp. $24.95 (December)

The Child's Theory of Mind

Henry M. Wellman

This book integrates the diverse strands of this rapidly expanding field of study. *The Child's Theory of Mind* charts children's knowledge about the mind, and characterizes that developing knowledge as a coherent commonsense theory.

400 pp. $35.00 (July)

Cognitive Models of Speech Processing

Psycholinguistic and Computational Perspectives

edited by Gerry T. M. Altmann

Cognitive Models of Speech Processing presents extensive reviews of current thinking on psycholinguistic and computational topics in speech recognition and natural language processing, along with a substantial body of new experimental data and computational simulations.

500 pp. $45.00 (December)

Laura

A Case for the Modularity of Language

Jeni E. Yamada

Possibly the most in-depth and comprehensive study of selectively intact language done to date. The case covered in this book counters claims that cognitive, social/interactive, and perceptual factors can wholly account for language acquisition and upholds the notion that language is a highly evolved, specialized human ability driven at least in part by a set of principles seen in no other cognitive domains.

100 pp. $22.50 (December)

Neurocomputing 2

Directions for Research

edited by James A. Anderson, Andras Pellionisz, and Edward Rosenfeld

The first volume of *Neurocomputing* has become an established guide to the background of concepts employed in this burgeoning field. *Neurocomputing 2* collects 39 articles covering network architecture, neurobiological computation, statistics and pattern classification, and problems and applications that suggest important directions for the evolution of neurocomputing.

900 pp. $60.00 (October)

Now Available in Paperback

Neurocomputing

Foundations of Research

edited by James A. Anderson and Edward Rosenfeld

729 pp. $30.00 paper

Neural Networks for Control

edited by W. Thomas Miller, III, Richard S. Sutton, and Paul J. Werbos

This book highlights key issues in learning control and identifies research directions that could lead to practical solutions for control problems in critical application domains.

500 pp. $45.00 (October)

Original in Paperback

Neurobiology of Cognition

edited by Peter D. Eimas and Albert M. Galaburda

The six essays in this book focus on the empirically answerable issue of whether and to what extent it is possible to explain observations about the mind in terms of observations about the brain.

256 pp. $22.50 paper (August)

Visit our booth or order toll-free: 1-800-356-0343 or (617) 625-8569. Fax orders: 617-625-6660. MasterCard and Visa accepted.

Modular Deficits in Alzheimer-Type Dementia

edited by Myrna F. Schwartz

These contributions present sound evidence that diseases of the Alzheimer type compromise brain function highly selectively, affecting some aspects of cognition while sparing others. Included are original case studies that explore in detail the nature of the linguistic, semantic, and visuoperceptual disorders in patients with degenerative dementias.

456 pp. $45.00 (November)

New Ideas on the Structure of the Nervous System in Man and Vertebrates

Santiago Ramón y Cajal
translated by Neely Swanson and Larry W. Swanson
foreword by Larry W. Swanson

New Ideas presents the histological evidence for the laws governing the form and connections of nerve cells. This work and the principles that emerged from it formed the cornerstone for our current understanding of how the nervous system is organized.

200 pp., illus. $29.95 (September)

About Time

Inventing the Fourth Dimension

William J. Friedman

A new integrated look at research on the psychological processes that underlie the human experience of time. 160 pp. $19.95 (August)

As Time Goes By

Tense and Universal Grammar

Norbert Hornstein

Hornstein proposes a theory of natural-language tense responsive to the language-acquisition problem.

256 pp. $25.00 (August)

Bring in an exhibit order form for a 20% discount at our bookstore in Kendall Square.

The MIT Press

55 Hayward Street
Cambridge, MA 02142

Outstanding new textbooks in cognitive psychology... ...only from LEA

A STUDENT'S HANDBOOK OF COGNITIVE PSYCHOLOGY Second Edition

Michael W. Eysenck
Bedford College, University of London
Mark Keane
University of Wales, Cardiff

"I think this is an excellent book which has a good chance of capturing the market as ***the*** *definitive text book on cognitive psychology. It is a marked advance on the previous 'Handbook:' interweaving the three approaches, experimental, neuropsychology and cognitive science, makes the book more interesting, more authoritative and distinguishes it from other texts."*

Gillian Cohen
The Open University

Regarded as a second edition of the extremely successful *Handbook of Cognitive Psychology*, this revision has been entirely rewritten and updated in response to the changes in cognitive psychology itself.

Traditional, experimental cognitive psychology is not supported by a cognitive science perspective that stresses computational modelling and a cognitive neuropsychology that points up the effects of brain damage on cognition. One of the major changes in this new *Handbook* is its treatment of the dynamic impact of these different views on the main topic areas of cognitive psychology including perception, attention, memory, categorization, language, problem solving, and reasoning. As such, it provides a very comprehensive account of the current state of cognitive psychology.

This text will be easily accessible to undergraduate students through the use of section summaries and point-by-point accounts of theories, and should be of value to students of psychology at all levels as well as those taking courses in computer science, physiology and medicine.

0-86377-153-X [cloth] / 1990 / 432pp. / $39.95
0-86377-154-8 [paper] / $18.50

VISUAL PERCEPTION Physiology, Psychology and Ecology Second Edition

Vicki Bruce
Patrick R. Green
University of Nottingham

"A very thorough updating. Will make the textbook even more useful than before. I think the book is the best on the market, by far, for second and third year undergraduates taking perception courses in psychology, zoology and visual science/optometry. Also undoubtedly provides good groundwork for post-graduates and researchers in the field of perception, animal behavior and computer vision."

Mark Georgeson
Bristol University

Recent theoretical developments and research findings from three different approaches to visual perception are brought together in this book. The first is physiological -- describing the evolution of different types of eyes and the physiology of mammalian visual pathways. The second is the traditional psychological position -- discussing perceptual organization, the perception of depth and motion, and pattern recognition in terms of the processing of information contained in retinal images -- with special emphasis placed on recent computational work on these processes. The third viewpoint is ecological, particularly as it is applied to problems of animal and human locomotion, event perception and understanding of the social world, including the human face.

This important book stands out from other texts on visual perception in treating both the computational and ecological approaches in detail, and in seeking to clarify the points of contact and controversy between them. It is also distinctive in placing emphasis on animal vision as well as that of humans.

Aimed at advanced undergraduate psychology and zoology students, this volume will also prove valuable to postgraduate students and research workers.

0-86377-145-9 [cloth] / 1990 / 348pp. / $49.95
0-86377-146-7 [paper] / $18.50

Call toll-free to order: 1-800-9-BOOKS-9
Lawrence Erlbaum Associates, Inc.
365 Broadway, Hillsdale, NJ 07642
201/666-4110 FAX 201/666-2394

The first volume in a new LEA series...
Studies in Cognition

The Adaptive Character of Thought

John R. Anderson
Carnegie Mellon University

series editors:
Allan Collins, BBN Laboratories
Edward E. Smith, University of Michigan
Kurt VanLehn, Carnegie Mellon University

The first publication in the *Studies in Cognition* series, this important volume examines the phenomena of cognition from an adaptive perspective. Instead of the typical cognitive psychology practice of trying to predict behavior from a model of cognitive mechanisms, *The Adaptive Character of Thought* develops a number of models that successfully predict behavior from the structure of the environment to which cognition is adapted. The methodology -- called rational analysis -- involves specifying the information-processing goals of the system, the structure of the environment, and the computational constraints on the system. This allows predictions about behavior to be derived by determining what behavior would be optimal under these assumptions.

The book applies the rational analysis methodology to these four cognitive phenomena:

- Memory -- showing that memory phenomena can be understood as optimal responses to the task of information retrieval in the presence of uncertain cues.
- Categorization -- showing that categorization behavior can be understood as an attempt to optimize the ability to predict features of new objects.
- Causal inference -- showing that causal judgments reflect an attempt to maximize prediction about the future given the scientific structure of the environment.
- Problem solving -- analyzed as an attempt to maximize the external goals achieved while minimizing both external and mental effort.

Contents: Part I: *Introduction.* Levels of a Cognitive Theory. Current Formulation of the Levels Issue. The New Theoretical Framework. Is Human Cognition Rational? The Rest of the Book. **Part II:** *Memory.* A Rational Analysis of Memory. The History Factor. The Contextual Factor. Relationship of Need Probability to Probability and Latency of Recall. Combining Information from Cues. Implementation in the ACT Framework. Effects of Subject Strategy. Conclusions. **Part III:** *Categorization.* The Goal of Categorization. The Structure of the Environment. The Optimal Solution. An Iterative Algorithm for Categorization. Application of the Algorithm. Survey of the Experimental Literature. Conclusion. **Part IV:** *Causal Inference.* Causal Estimation. Cues for Causal Inference. Integration of Statistical and Temporal Cues. Discrimination. Abstraction of Causal Laws. Implementation in a Production System. Conclusion. **Part V:** *Problem Solving.* Making a Choice Among Simple Actions. Combining Steps. Studies of Hillclimbing. Mean-Ends Analysis. Instantiation of Indefinite Objects. **Part VI:** *Conclusions on Rational Analysis.* Implementation in ACT.
0-8058-0419-6 (cloth) / June 1990 / approx. 304 pp. / $45.00
Special Prepaid Offer! $27.50
No further discounts apply.

Lawrence Erlbaum Associates, Inc.
365 Broadway, Hillsdale, NJ 07642
201/666-4110 FAX 201/666-2394

Call toll-free to order: 1-800-9-BOOKS-9.
Please have your credit card or LEA customer ID# at hand.

Annual Conferences of the Cognitive Science Society

Proceedings of the:

Eleventh	(1989)	$79.95
Tenth	(1988)	$79.95
Ninth	(1987)	$89.95
Eighth	(1986)	$89.95
Seventh	(1985)	$59.95
Sixth	(1984)	$49.95
Fifth	(1983)	$49.95
Fourth	(1982)	$49.95
Third	(1981)	$49.95

Available from...

Lawrence Erlbaum Associates, Inc.
365 Broadway, Hillsdale, NJ 07642
201/666-4110 FAX 201/666-2394

Call toll-free to order: 1-800-9-BOOKS-9
(1-800-926-6579)
Please have your credit card or
LEA customer ID number ready.

ISSN 1047-1316

ISBN 0-8058-0938-4