The Tenth Annual Conference of the Cognitive Science Society

August 17-19, 1988

Montreal, Quebec, Canada

Program of the

Tenth Annual Conference of the Cognitive Science Society

17-19 August 1988

Montreal, Quebec, Canada

LEA LAWRENCE ERLBAUM ASSOCIATES, PUBLISHERS
1988 Hillsdale, New Jersey Hove and London

Copyright 1988 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 any other means without the prior written permission of the publisher.

Distributed by

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

ISBN 0-8058-0436-6

Printed in the United States of America

The Tenth Annual Conference of the Cognitive Science Society

La Dixième Conférence Annuelle de la Société de Science Cognitive

Vimla L. Patel
Guy J. Groen

Conference Co-Chairs

COMMITTEES

Scientific Program Committee

Mark Seidenberg, Thomas Shultz — co-chairs

David Caplan
David Evans
Guy Groen
Serge Larochelle
Vimla Patel

Local Arrangements

Lorence Coughlin, David Kaufman — co-chairs

José Arocha
Aldo Braccio
Stephen Chase
Anoop Chawla
Micheal Hoover
Debra Jarryd
Hélène Pineau
André Renaud
Daniel Saumier

REVIEWERS

Robert Bracewell
Daniel Bub
David Caplan
Patrick Cavanaugh
Kevin Dunbar
David Evans
Carl Frederiksen
Guy Groen
Serge Larochelle
A. AJ Marley
Vimla Patel
Franz Schmalhofer
Mark Seidenberg
Thomas Shultz
Rand Spiro

GOVERNING BOARD

John R. Anderson, chair Elect	Aug. 1983 -- Aug. 1989
Joan Bresnan	Aug. 1983 -- Aug. 1989
Ann Brown	Aug. 1984 -- Aug. 1990
Dedre Gentner	Aug. 1987 -- Aug. 1993
James Greeno	Aug. 1987 -- Aug. 1993
Patrick J. Hayes	Aug. 1984 -- Aug. 1990
Geoffrey Hinton	Aug. 1986 -- Aug. 1992
Walter Kintsch	Aug. 1988 -- Aug. 1994
Jay McClelland	Aug. 1988 -- Aug. 1994
Ulric Neisser	Aug. 1985 -- Aug. 1991
Donald A. Norman	Aug. 1985 -- Aug. 1991
David Rumelhart	Aug. 1986 -- Aug. 1992

EX OFFICIO BOARD MEMBERS

Gordon Bower, Chair	Aug. 1987 -- Aug. 1988
Martin Ringle, Executive Editor of Cognitive Science	Aug. 1986 --
Kurt VanLehn, Secretary/Treasurer	Aug. 1985 -- Aug. 1988

This conference is supported by grants from
The James McDonnell Foundation
and
The Faculties of Medicine and Graduate Studies of
McGill University.

Proceedings of the
10th Annual Conference of the Cognitive Science Society
Montreal, Canada - August 17 - 19, 1988

PAPER PRESENTATIONS

The Place of Cognitive Architectures in a Rational Analysis 1
John Anderson

Transitions in Strategy Choices 11
Robert Siegler

VITAL, A Connectionist Parser 18
Tim Howells

Applying Contextual Constraints in Sentence Comprehension 26
Mark St. John
J. L. McClelland

Recursive Auto-Associative Memory: Devising Compositional Distributed Representations 33
Jordan Pollack

Experiments with Sequential Associative Memories 40
Stephen I. Gallant
D. J. King

Representing Part-Whole Hierarchies in Connectionist Networks 48
Geoffrey Hinton

Using Rules and Task Division to Augment Connectionist Learning 55
William L. Oliver
Walter Schneider

Analyzing a Connectionist Model as a System of Soft Rules 62
C. McMillan
Paul Smolensky

How to Summarize Text (and Represent it too) 69
R. Alterman
Lawrence A. Bookman

On-Line Processing of a Procedural Text 76
Andre Renaud
C. H. Frederiksen

Understanding Stories in their Social Context 83
Eric Domeshek

Context Effects in the Comprehension of Idioms 90
Patrizia Tabossi
C. Cacciari

Action Planning: Routine Computing Tasks 97
Suzanne M. Mannes
W. Kintsch

Using Conversation MOPs to Integrate Intention and Convention in Natural Language Processing 104
Elise H. Turner
R. E. Cullingford

A Theory of Simplicity 111
Gilbert Harman
M. Ranney
K. Salem
F. Doring
J. Epstein
A. Jaworska

Basic Levels in Hierarchically Structured Categories 118
James E. Corter
M. A. Gluck
G H. Bower

Flexible Natural Language Processing and Roschian Category Theory 125
Sandra Peters
S. C. Shapiro
W. J. Rapaport

The Induction of Mental Structures While Learning to Use Symbolic Systems 132
Thomas G. Bever
R. E. Hansen

Transitory Stages in the Development of Medical Expertise: The "Intermediate Effect" in Clinical Case Representation Studies 139
Henk Schmidt
H. P. A. Boshuizen
P. P. M. Hobus

Integrating Marker Passing and Connectionism for Handling Conceptual and Structural Ambiguities 146
Ronald A. Sumida
M. G. Dyer
M. Flowers

Learning Subgoals and Methods for Solving Problems 153
Richard Catrambone
K. Holyoak

Opportunistic Use of Schemata for Medical Diagnosis 160
Roy Turner

Integrating Case-Based and Causal Reasoning 167
Phyllis Koton

Modeling Software Design Within a Problem-Space Architecture 174
Beth Adelson

Modeling Human Syllogistic Reasoning in Soar 181
Thad A. Polk
A. Newell

Integrated Commonsense and Theoretical Mental Models in Physics Problem Solving 188
Jeremy Roschelle

A Connectionist Model of Selective Attention in Visual Perception 195
Michael C. Mozer

How Near is too Far? Talking About Visual Images 202
U Zernik
B. J. Vivier

An Adaptive Model for Viewpoint - Invariant Object Recognition 209
Peter Sandon
L. M. Uhr

Spatial Reasoning Using Sinusoidal Oscillations 216
Ian Pratt

An Unsupervised PDP Learning Model for Action Planning 223
Yoshiro Miyata

Representation and Recognition of Biological Motion 230
Nigel Hugh Goddard

When half right is not half bad: Hypothesis testing under conditions of uncertainty and complexity 237
Joshua Klayman

A Theory of Scientific Problem Solving 244
Randolph Jones
Pat Langley

Empirical Analyses and Connectionist Modeling of Real-Time Human Image Understanding 251
I. Biederman
T.W. Blickle
G. Ju
H.J. Hilton
J.E. Hummel

Learning to Represent and Understand Locative Prepositional Phrases 257
C. Cosic
Paul Munro

A Computational Model of Syntactic Ambiguity as a Lexical Process 263
Curt Burgess
S. C. Hollbach

A Parallel Model for Adult Sentence Processing 270
Robin Clark
E. Gibson

Parsing Metacommunication in Natural Language Dialogue to Understand Indirect Requests 277
David Sanford
J. W. Roach

Interpretation of Quantifier Scope Ambiguities 284
Howard S. Kurtzman
M. C. MacDonald

Multiple Simultaneous Interpretations of Ambiguous Sentences 291
Peter Norvig

The Role of Analogy in a Theory of Problem Solving 298
Beth Adelson
Dedre Gentner
K. Hammond
Keith Holyoak
Paul Thagard

The Architecture of Children's Physics Knowledge: A Problem-Solving Perspective 305
Kathleen E. Metz

Hierarchical Problem Solving as a Means of Promoting Expertise 312
Jose Mestre
R. Dufresne
W. Gerace
P. Hardiman

Collaborative Cognition 319
Barbara Fox
L. Karen

Explorations in Understanding How Physical Systems Work 325
Barbara Y. White
J. R. Frederiksen

Instructional Strategies for a Coached Practice Environment 332
Susanne P. Lajoie
G. M. Eggan
A. M. Lesgold

Creatures of Habit: A Computational System to Enhance and Illuminate the Development of Scientific Thinking 340
Roy D. Pea
M. Eisenberg
F. Turbak

Propositional Attitudes, Commonsense Reasoning and Metaphor 347
John A. Barnden

A Process-Oriented, Intensional Model of Knowledge and Belief 354
Robert F. Hadley

Subcognitive Probing: Hard Questions for the Turing Test 361
Robert M. French

The Pragmatics of Expertise in Medicine 368
David A. Evans

Cognitive Flexibility Theory: Advanced Knowledge Acquisition in Ill-Structured Domains 375
Rand J. Spiro
R. L. Coulson
Paul J. Feltovich
D. Anderson

A Hybrid Connectionist/Production System Interpretation of Age Differences in Perceptual Learning 384
A.D. Fisk
W.A. Rogers

Language Experience and Prose Processing in Adulthood 391
Roger A. Dixon
Lars Backman

Effects of Age and Skill on Domain-Specific Visual Search 398
S.M. Clancey
W.J. Hoyer

Patching Up Old Plans 405
Katia Sycara

Systematicity as a Selection Constraint in Analogical Mapping 412
Catherine Clement
D. Gentner

Abstraction Processes During Concept Learning: A Structural View 419
Janice Skorstad
D. Medin
D. Gentner

Explanatory Coherence and Belief Revision in Naive Physics 426
Michael Ranney
P. Thagard

Access and Use of Previous Solutions in a Problem Solving Situation 433
Jeremiah M. Faries
B. J. Reiser

The Use of Explanations for Completing and Correcting Causal Models 440
Joel D. Martin
M. Redmond

Varieties of Learning from Problem Solving Experience 447
J. Lancaster
Janet L. Kolodner

The Process of Learning LISP 454
Frederick G. Conrad
J. R. Anderson

Interactive Medical Problem Solving in the Context of the Clinical Interview: The Nature of Expertise 461
David Kaufman
Vimla Patel

A Dynamical Theory of the Power-Law of Learning in Problem-Solving 468
Jeffrey C. Shrager
T. Hogg
B. A. Huberman

Assessing the Structure of Knowledge in a Procedural Domain 475
Michel Desmarais
L. Giroux
S. Larochelle
S. Leclerc

Improvement in Medical Knowledge Unrelated to Stable Knowledge 482
Geoff Norman
L. Brooks
S.W. Allen
D. Rosenthal

POSTER PRESENTATIONS

Sequential Connectionist Networks for Answering Simple Questions about a Microworld 489
Robert B. Allen

The Usefulness of the Script Concept for Characterizing Dream Reports 496
D. Deslauriers
George W. Baylor

The Right of Free Association: Relative-Position Encoding for Connectionist Data Structures 503
John A. Barnden

Unsupervised Learning of Correlational Structure 510
A. Chalnick
Dorrit O. Billman

On the Application of Medical Basic Science Knowledge in Clinical Reasoning: Implications for Structural Knowledge Differences Between Experts and Novices 517
Henny Boshuizen
H. G. Schmidt
L. D. Coughlin

Similarity-Based and Explanation-Based Learning of Explanatory and Nonexplanatory Information 524
W. K. Ahn
William F. Brewer

Causal Reasoning about Complex Physiological Mechanisms by Novices 531
Anoop Chawla
Vimla Patel

A Comparison of Context Effects for Typicality and Category Membership Ratings 538
Leslie J. Caplan
R. A. Barr

Naive Materialistic Belief: An Underlying Epistemological Commitment 544
M. Reiner
Micki Chi
L. B. Resnick

Writing Expertise and Second Language Proficiency: Algorithms and Implementations? 552
Alister Cumming

The Minimal Chain Principle: A Cross Linguistic Study of Syntactic Parsing Strategies 559
Marica De Vincenzi

Multiple Character Recognition - A Simulation Model 566
S. Koebe
Gerhard Deffner

NETZSPRECH - Another Case for Distributed `Rule' Systems 573
Georg Dorffner

Intuitive Notions of Light and Learning About Light 580
M. Reiner
Menahem Finegold

Signalling Importance in Spoken Narratives: The Cataphoric Use of the Indefinit THIS 587
Morton A. Gernsbacher
S. Shroyer

Planning and Implementation Errors in Algorithm Design 594
Wayne D. Gray
A. T. Corbett
K. Van Lehn

Conceptual Slippage and Analogy-Making: A Report on the Copycat Project 601
Douglas R. Hofstadter
M. Mitchell

Direct Inferences in a Connectionist Knowledge Structure 608
Susan C. Hollbach

Problem Solving is what you do when you don't know what to do 615
Paul E. Johnson
L. K. Kochevar

Cirrus: Inducing Subject Models from Protocol Data 623
K. Van Lehn
Bernadette Kowalski

Three Kinds of Concepts? 630
Barbara Malt

Constructing Coherent Text Using Rhetorical Relations 637
Johanna Moore
C. L. Paris

Defeasibility in Concept Combination: A Critical Approach 644
Terry Myers
B. Franks
S. McGlashan

The Comprehension of Architectural Plans By Expert and Sub-Expert Architects 651
J.D. Gobert
C.H. Frederiksen

Processing Aspectual Semantics 658
Sergei Nirenburg
J. Pustejovsky

Gain Variation in Recurrent Error Propagation Networks 666
Steven J. Nowlan

Conjoint Syntactic and Semantic Context Effects: Tasks and Representations 673
Padraig O'Seaghdha

Generalization by Humans and Multi-Layer Adaptive Networks 680
Michael Pavel
M. A. Gluck
V. Henkle

Pattern-Based Parsing Word-Sense Disambiguation 688
R. S. Virkar
John W. Roach

Multiple Theories in Scientific Discovery 695
Donald Rose

Text Comprehension: Macrostructure and Frame 702
Jean Pierre Rossi

The Structure of Social Mind: Empirical Patterns of
Large-scale Knowledge Organization 710
David L. Sallach

A Model of Meter Perception in Music 717
B. O. Miller
Don L. Scarborough
J. A. Jones

Acquiring Computer Skills by Exploration versus
Demonstration 724
Franz Schmalhofer
O. Kühn

A Hybrid Model for Controlling Retrieval of Episodes 731
Colleen M. Seifert

The Role of Mapping in Analogical Transfer 738
Hong Shinn

Spatial Attention and Subitizing: An Investigation of the
FINST Hypothesis 745
Lana Trick
Z. Pylyshyn

A Computational Model of Reactive Depression 758
Charles Webster
R. M. Glass
G. Banks

Reasoning by Rule or Model ?.....765
P.N. Johnson-Laird

The Place of Cognitive Architectures in a Rational Analysis

John R. Anderson

Department of Psychology

Carnegie-Mellon University

This paper contains a summary of the main points that I will be making in my presentation at the Cognitive Science Meetings. Some more details will be in that presentation. All the details and formal derivations will be found in Anderson (in press).

This paper will consider the Soar architecture of Laird, Newell, and Rosenbloom (in press), my own ACT* architecture (Anderson, 1983), and the PDP architecture of McClelland and Rumelhart (Rumelhart & McClelland, 1986, McClelland & Rumelhart, 1986). Now that there are numerous candidates for cognitive architectures, one is naturally led to ask which might be the correct one or the most correct one. This is a particularly difficult question to answer because these architectures are often quite removed from the empirical phenomena which they are supposed to account for. In actual practice one sees proponents of a particular architecture arguing for that architecture by reference to what I call signature phenomena. These are empirical phenomena which are particularly clear manifestations of the purported underlying mechanisms. The claim is made that the architecture provides particularly natural accounts for these phenomena and that these phenomena are hard to account for in other architectures. In this paper I will argue that the purported signature phenomena tell us very little about what is inside the human head. Rather they tell us a lot about the world in which the human lives. The majority of this paper will be devoted to making this point with respect to examples from the SOAR, ACT*, and PDP architecture.

As a theorist who has been associated with the development of cognitive architectures for 15 years I should say a little about how I came to be advocating this position. I have been strongly influenced by David Marr's (1982) metatheoretical arguments in his book on vision which are nicely summarized in the following quote:

> An algorithm is likely to be understood more readily by understanding the nature of the problem being solved than by examing the mechanism (and the hardware) in which it is solved.

Marr made this point with respect to phenomena such as stereopsis where he argued that one will come to an understanding of the phenomena by focusing on the problem of how two two-dimensional views of the world contained enough information to enable one to extract a three-dimensional interpretation of the world and not by focusing on the mechanisms of stereopsis. He thought his viewpoint was appropriate to higher-level cognition although he did not develop it for that application. As recent as a few years ago I could not see how his viewpoint applied to higher level cognition (Anderson, 1987). However, in

the last couple of years I have come to see how it would apply and have realized its advantages. The basic point of Marr's was that if there is an optimal way to use the information at hand the system will use it. I have stated this as the following principle:

Principle of Rationality. The cognitive system operates at all times to optimize the adaptation of the behavior of the organism.

One can regard this principle as being handed to us from outside of psychology--as a consequence of basic evolutionary principles. However, I do not want to endorse this viewpoint on that principle because there are many cases where evolution does not optimize. Rather, I view it as an empirical hypothesis to be judged by how well theories that embody the principle of rationality do in predicting various cognitive phenomena. Developing a theory in a rational framework involves the following 6 steps:

1. Precisely specify what the goals of the cognitive system are.

2. Develop a formal model of the environment that the system is adapted to (almost certainly less structured than the experimental situation).

3. Make the minimal assumptions about computational costs.

4. Derive the optimal behavioral function given (1)-(3).

5. Examine the empirical literatures to see if the predictions of the behavioral function are confirmed.

6. If predictions are off, iterate.

The theory in a rational approach resides in the assumptions in (1) - (3) from which the predictions flow. I refer to these assumptions as the framing of the information processing problem. Note this is a mechanism-free casting of a psychological theory. It can be largely cast in terms of what is outside of the human head rather than inside. As such it enjoys another advantage which is that its assumptions are potentially capable of independent verification.

SOAR--Power Law Learning

The signature phenomenon I would like to consider for the SOAR theory is power-law learning which is referenced in many of the SOAR publications. This refers to the linear relationship that is obtained between the logarithm of the amount of practice and the logarithm of response time which implies that the performance measure is a power function of practice. In the Soar model the power law falls out of the chunking learning mechanism plus some critical auxiliary assumptions. Chunking refers to the collapsing of multiple production firings into a single production firing that does the work of the set. It is assumed that each chunk produces a performance enhancement proportional to the number

of productions eliminated. Chunks are formed at a constant rate--either on every opportunity or with equal probability on every opportunity. The final critical assumption is that as chunks span larger and larger units the number of potential chunks grows exponentially. As a consequence of the last assumption, learning will progress ever more slowly because it takes more experience to encounter all of the larger chunks.

I will offer a rational analysis of power law learning which will also explain the forgetting and massing functions. This will be part of a larger rational analysis of human memory which is the topic of the next section.

A Rational Analysis of Human Memory

The claim that human memory is rationally designed might strike one at least as implausible as the general claim for the rationality of human cognition. Human memory is always disparaged in comparison to computer memory--it is thought of as slow both in storage and retrieval and terribly unreliable. However, such analyses of human memory fail both to understand the task faced by human memory and the goals of memory. I think human memory should be compared with information-retrieval systems such as the ones that exist in computer science. According to Salton and McGill (1983) a generic information retrieval system consists of four things:

(1) There is a data base of files such as book entries in a library system. In the human case these files are the various memories of things past.

(2) The files are indexed by terms. In a library system the indexing terms might be keywords in the book's abstract. In the human case the terms are presumably the concepts and elements united in the memory. Thus, if the memory is seeing Willie Stargell hit a home run the indexing terms might be Willie Stargell, home run, Three Rivers Stadium, etc.

(3) An information retrieval system is posed queries consisting of terms. In a library system these are suggested keywords by the user. In the case of the human situation it is whatever cues are presented by the environment such as when someone says to me "Think of a home run at Three Rivers Stadium".

(4) Finally there are a set of target files desired by which we can judge the success of the information retrieval.

One thing that is very clear in the literature on information retrieval systems is that they cannot know the right files to retrieve given a query. This is because the information in the queries does not completely determine what file is wanted. The best information retrieval systems can do is assign probabilities to various files given the query. Let us denote the probability that a particular file is a target by P[A]. In deciding what to do informational retrieval systems have to balance two costs. One is what Salton an McGill call the precision cost and which I will denote C_P. This is the cost associated with retrieving a file which is not a target. There must be a corresponding cost in the human system. This is the one place where we will see a computational cost appearing in our rational analysis of

memory. The other cost Salton and McGill call the recall cost and we will denote it C_R. It is the cost associated with failing to retrieve a target. Presumably in most cases it is much larger than the precision cost for a single file or memory.

Given this framing of the information processing problem we can now proceed to specify the optimal information-processing behavior. This is to consider memories (or files) in order of descending P[A] and stop when the expected cost associated with failing to consider the next item is greater than the cost associated with considering it or when

$$P[A]\, C_R < (1\text{-}P[A])\, C_P \tag{1}$$

We now have a complete theory of human memory except for one major issue--how should the system go about estimating P[A]. I propose that the system should use the item's past history of usage and the elements in the current context to come up with a Bayesian estimate of that probability. A particularly transparent way of stating this is with the Bayesian odds ration formula which we can state

$$\frac{P(A|H_A\&Q)}{P(\overline{A}|H_A\&Q)} = \frac{P(A|H_A)}{P(\overline{A}|H_A)} \cdot \prod_{i \in Q} \frac{P(i|A)}{P(i|\overline{A})} \tag{2}$$

where $P(A|H_A\&Q)$ is the posterior probability that the memory is needed given its past history and the cues in the current context, $P(\overline{A}|H_A\&Q)$ is $1\text{-}P(A|H_A\&Q)$, $P(A|H_A)$ is the posterior probability given just the history, $P(\overline{A}|H_A) = 1\text{-}P(A|H_A)$, $P(i|A)$ is the conditional probability that i would be in the current context if A is needed, and $P(i|\overline{A})$ is the conditional probability if A is not needed. This way of formulating the relationship nicely breaks up the need probability into the product of a history factor $P(A|H_A)/P(\overline{A}|H_A)$ plus a context factor the product involving the $P(i|A)/P(i/\overline{A})$. Note that in this context factor we are assuming the individual cues are independent of one another in order to obtain a product. I neither want to argue that this is really true nor that the human system actually acts as if it is. I am only using this as an approximation to get an indication of what the rational predictions are.

The History Factor

In investigating the implications of this rational analysis for the power-law learning function we need to focus on the history factor in the above equation. In particular we need to specify $P(A|H_A)$. To determine this we need to know how the past history of usage of a memory trace predicts whether it will be currently used. To determine this in a truly valid objective way we would have to follow people around, determine when they use particular facts, and induce what the empirical relationship is. It is nearly impossible to imagine collecting such objective statistics in the human case but such statististics are available for other information retrieval systems. For instance, there is data about how past borrowings from a library predict future borrowings(Burrell, 1980; Burrell & Cane, 1982). There is data

about how past accesses to a file predict future accesses(Stritter, 1977). The data for these different domains is quite similar in terms of the nature of the functional relationship between past use and current use. I propose that these relationships are true of all information retrieval systems including the human one.

Burrell developed a model for library borrowings which provides a good analytical starting point. There are three basic assumptions in Burrell's model. The first is that the items in a system vary in their desirability. Burrell assumes that the distribution of desirability is a gamma distribution with parameter b and index v. He is able to basically show such a distribution of borrowings in the case of a library system. The second assumption that Burrell makes is that there is an aging process such that items will decay in their borrowing rate with the passage of time. Again he can empirically validate that such an aging process does occur. This means that if we take an item from the gamma distribution with initial desirability λ its desirability after time t will be $\lambda r(t)$ where r(t) describes the rate of decay. Burrell uses a simple exponential decay in rate of the form. The third assumption of Burrell is that borrowings are a Poisson process and that times until next borrowing are exponentially distributed with rate $\lambda r(t)$.

With these assumptions we can derive what I call the recency-frequency function RF(n,t) which is the probability that an item introduced t time units ago and used n times over that period will be needed in the current time unit. It produces a linear relationship between number of uses, n, and need probability. This is a special case of a power function. When we consider plausible monotonic transformations from need probability to latency the linear relationship disappears but the power function relationship remains. Because of the aging factor r(t) we wind up predicting the forgetting function quite accurately as well.

Thus, we have shown that power law learning can be predicted from a rational perspective which sees human memory as adapting to the statistics of information use. Thus, it is what is outside the human head not what is inside that is controlling the memory performance. I should emphasize that this does not deny that chunking may be one of the mechanisms the mind uses to achieve this adaption. However, the argument is that the real explanation is in the outside world and not in the internal mechanisms.

ACT*--The Fan Effect

Now I would like to turn to the second architecture, ACT*, and consider a signature phenomenon which has played a key role in its development. This is the fan effect (Anderson,1983) . A typical experiment is focused on subjects' ability to recognize sentences that they have learned. According to ACT*, upon being presented with a sentence such as "The lawyer is in the park" the subject activates the concepts in the sentence such as lawyer, in, and park. Activation spreads from these concepts along various network paths. The time to recognize a sentence is a function of the amount of activation reaching the proposition node. The critical additional assumption in the ACT* theory is that the amount of activation that can spread out of a node is fixed and that the more paths emanating out of a concept the less activation can go to any one proposition and so the slower recognition will be. Fan refers to the number of such paths and is manipulated by manipulating the number of facts studied about a concept like lawyer.

We can extend our previous rational analysis of the fan fan effect to accomodate the fan effect. Here we will be interested in analyzing the context factor rather than the history factor since we are manipulating properties of the memory cues that we presented to subjects. That is we want to focus on the quantities $P(i|A)/P(i|\overline{A})$ where the i are concepts like lawyer and the A are the sentences to be recalled. We can rewrite these as

$$\frac{P(i|A)}{P(i|\overline{A})} = \frac{P(A|i)P(i)/P(A)}{P(\overline{A}|i)P(i)/P(\overline{A})} \tag{3}$$

The P(i) drop out. Since $P(\overline{A})$ must be near one (there are millions of traces and no one can be very probable) it can also be ignored. To an approximation we can also ignore $P(\overline{A}|i)$. This is a good approximation to the extent that the probability of needing a trace remains low even in the presence of a predictive cue. If we allow this approximation we get the following which is very easy to analyze:

$$\frac{P(i|A)}{P(i|\overline{A})} \simeq \frac{P(A|i)}{P(A)} \tag{4}$$

Our claims do not depend on making this approximation. It is just that they are a lot easier to see with the approximation. In our experiments P(A) is basically constant for all items and so the critical factor turns out to be the probability that the trace is relevent given a particular cue. This is precisely what is manipulated by fan in a typical experiment. The more facts associated with a particular concept the less likely any one is given the concept. Basically if the fan is n the probability is 1/n. Anderson (1976) did an experiment that decorrelated fan and probability by manipulating the probability of testing various facts associated with a particular concept. That experiment showed conclusively that the critical factor is probability and not fan.

Thus, the fan effect is a consequence of memory using the correlation between cues and a memory's relevence to predict when the memory is needed. It may be that spreading activation is one of the mechanisms that the mind uses to compute the correlation. However, for current purposes the critical fact is once again that the explanation of the phenomena lies in what is outside of the human head and not what is inside.

PDP -- Categorization

PDP models involve representing knowledge in a distributed form where specific experiences do not have specific encodings. On the other hand PDP models do learning locally such that changes in strengths of connection between specific elements must underlie these distributed encodings. This leads PDP models to naturally produce generalization phenomena such that they extract central tendencies out of the experience of specific instances. In introducing PDP models, McClelland, Rumelhart, & Hinton(1986) give a lot of play to categorization phenomena which is the identification of common categories in a set of tendencies. It receives more page space in their article than any other phenomena. There is a substantial literature in cognitive psychology on categorization behavior.

McClelland et al. do not actually simulate any specific experiment in this literature but rather offer a simulation of the extraction of the characteristics of the members of two gangs (the jets and the sharks) as a prototype of the experiments in the literature.

To develop a rational analysis of categorization behavior the first thing we need to ask is what are the goals of the cognitive system in forming categories. In much of the experimental literature on categorization one gets the feeling that the driving force behind categorization is some sort of social conformity--that we need to learn to use the same labels to describe objects as do other people. However, this clearly cannot be all of the picture, particularly because people can learn to identify categories in the absence of any labels. I think the real function of categorization is to maximize the system's ability to predict properties of objects including their labels. Clearly, a system that can make accurate predictions will be in a position to maximize its goals.

The reason people form categories to maximize prediction is because of the nature of objects in the external world. Formally, the following is the characterization that I will assume in my rational derivations. I will assume that the world seen so far has consisted of n objects which are partitioned into s disjoint sets or categories. Each object can be classified according to some r dimensions (for simplicity I will only consider cardinal dimensions) where each dimension i has some m_i values. The members of a category belong in that category by virtue of possessing theoretical probabilities p_{ij} that they will display value j on dimension i. These probabilities provide the intensional definition of a category in contrast to its extensional definition which can be gotten simply by listing the category members.

These assumptions are intended as descriptions of the external world not just of the perception of the world in the human head. One can ask why the objects in the world should partition themselves in disjoint partitions defined by conjunctions of features. I cannot say I know the total answer but there are some obvious things to point at. For instance there is the genetic phenomenon of species which enforces a disjoint (no crossbreeding) partitioning of conjunctively defined categories (the common genetic code within a species). Other types of objects like physical elements and tools tend to produce similar disjoint partitionings of conjunctively defined categories. One can also question the probabilistic definition of category membership since this is in contradiction to the tradition in the artificial intelligence work on categories. However, I think it is indisputable that category members do display their features with only certain probabilities. Most labradors are black and have four legs but neither feature is displayed universally.

From these assumptions one can derive a Bayesian algorithm to assign objects to categories and to estimate the theoretical probabilities p_{ij}. Again, I do not have the space to go into the details of the algorithm. I have applied the algorithm to the now classic data of Medin and Schaffer (1978) where it did better than their original model using only a single parameter rather than their many. I have also applied it to the long series of experiments involving the Posner and Keele (1968) stimuli using an encoding of these materials developed by Hintzman (1986). It accounts for all the phenomena that Hintzman lists for these materials. I have also successfully predicted the results of a complicated experiment of Elio and Anderson (1981) which no model before Hintzman's was able to

account for. Rather than discussing the specific experiments in detail it is worthwhile listing some of the major phenomena that are known about human categorization and explaining how the model accounts for each:

1. Clearly the research indicates that, to a degree, people extract the central tendency of a set of instances in that their behavior is a function of the distance from that central tendency. This simply reflects a sensitivity to the statistical correlation between features and category identity which amounts to using conditional probabilities in a Bayesian analysis.

2. In addition to distance from a central tendency the literature has found an effect of distance from specific examples(eg., Medin & Schaffer, 1978). This is produced by the tendency of the model to break diverse categories into subcategories where the features cluster together. The reason for this is that predictive power is gained by such decomposition.

3. It has shown that when a category has multiple central tendencies subjects can pick this up (Neumann, 1977). As with point (2) this is produced by the tendency to break a large diverse category into smaller categories that increase predictability.

4. There is an effect of category size as was discussed with respect to the Posner & Keele task. This is simply a sensitivity to base rates.

5. Rosch, Mervis, Gray, Johnson, Boyes-Braem (1976) has documented the many circumstances in which there appear to basic level categories. The existance of such categories in our framework is simply a consequence of the fact that these categories maximize the predictability of the world--which is basically Rosch's original point.

6. It is not necessary for feedback on category membership to be given in order for categories to emerge(Fried and Holyoak, 1984). Categories will emerge any time they increase in predictability of the universe. However, by applying category labels we increase the amount of structure that can be predicted and so enhance the value of category membership. So, labels should enhance categorization but are not essential.

7. The more things that can be predicted from category membership the more likely a category is to be formed even though this means one has to learn more about a category (Billman, 1983).

Thus it seems that categorization phenomena can be again explained from a rational perspective assuming that the controlling factor is the structure of the world and not the structure in the human head. Note again this analysis does not deny that PDP mechanisms may be the way that the mind implements this rational analysis. However, it denies that PDP models provide an adequate explanation of the phenomena.

Conclusions

In summary we have looked at three cognitive architectures For each we have have taken a signature phenomenon and developed a reasonable model of the world in which that phenomenon occurs and the goals of humans operating in that world. We have made a few assumptions about computational costs which are not at all mechanism specific We have derived the signature phenomena as solutions to the optimization problems we defined. In each case this rational analysis led to an account that was as accurate or more accurate than the original mechanistic account.

Now we come to the hard question of what the implications are of these demonstrations. I am not really sure what the implications are but I will hazard a guess. This is that cognitive architectures should be viewed as notations for expressing the behavioral functions that emerge as the solutions to the optimization problems in a rational analysis. The real theory lies in the assumptions made in the statement of the optimization problem--i.e., the assumptions about the goals, the world, and the computational limitations. These assumptions do not have the same identifiability problems that the mechanistic models do and lead to a much deeper explanation of the phenomena at hand. However, something computationally powerful like a Turing-equivalent architecture is necessary if we are going to be able to express the solution to these optimization problems.

Thus the theory is in the framing of the information processing problem and the architectures provide notation for expressing the solutions to the optimization problems. I see a one-to-many mapping between framings and architectures. That is, one can take a single framing and for every architecture find some configuration of its mechanisms that enable the optimal behavior to be computed. Choice among architectures is then not to be determined by veracity of empirical predictions. Rather it is to be determined by how easy it is to work out the optimal behavior in that architecture. Ease of use is the classic criterion for selecting among notations. Empirical veracity is reserved for theories.

References

Anderson, J. R. (1976). *Language, memory, and thought.* Hillsdale, NJ: Erlbaum.

Anderson, J.R. (1983). *The Architecture of Cognition.* Cambridge, MA: Harvard University Press.

Anderson, J.R. (1983). Retrieval of information from long-term memory. *Science, 220,* 25-30.

Anderson, J. R. (1987). Methodologies for studying human knowledge. *The Behavioral and Brain Sciences, 10,* 467-505.

Anderson, J. R. (in press). *The Adaptive Character of Thought.* Hillsdale, NJ: Erlbaum.

Billman, D. (1983). *Inductive learning of syntactic categories.* Doctoral dissertation, University of Michigan, Ph.D. dissertation

Burrell, Q. L. (1980) A simple stochastic model for library loans. *Journal of Documentation, 36,* 115-132.

Burrell, Q. L (1985). A note on aging on a library circulation model. *Journal of Documentation, 41*, 100-115.

Burrell, Q L. & Cane V. R. (1982). The analysis of library data. *Journal of the Royal Statistical Society, Series A*(145), 439-471.

Elio, R. & Anderson, J. R. (1981). The effects of category generalizations and instance similarity on schema abstraction. *Journal of Experimental Psychology: Human Learning and Memory, 7*, 397-417

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.

Hintzman, D. L. (1986). Schema Abstraction in a Multiple-Trace Memory Model. *Psychological Review, 93*, 411-428.

Marr, D. (1982). *Vision.* San Francisco: Freeman.

McClelland, J. L., Rumelhart, D. E., and the PDP research group. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition.* Cambridge, MA: Bradford Books.

McClelland, J. L., Rumelhart D. E., & Hinton, G. E. (1986). *Parallel Distributed Processing.* Vol. 1: *The appeal of parallel distributed processing.* In D. E. Rumelhart & J. L. McClelland (Eds.).

Medin, D. L., & Schaffer, M. M. (1978). Context theory of classification learning. *Psychological Review, 85*, 207-238.

Neumann, P G. (1977). Visual prototype information with discontinuous representation of dimensions of variability. *Memory & Cognition, 5*, 187-197.

Posner, M. I. & Keele, S. W. (1968). On the genesis of abstract ideas. *Journal of Experimental Psychology, 77*, 353-363.

Rosch, E Mervis, C B., Gray W., Johnson D., & Boyes-Braem, P (1976) Basic objects in natural categories *Cognitive Psychology, 7*, 573-605.

Rosenbloom, P S. Laird, J. E. & Newell, A. (In Press). *Working Models of Human Perception. The chunking of skill in knowledge.* London: Academic Press. In H. Buoma & B. A. G. Elsendoorn (Eds.).

Rumelhart, D. E. McClelland J L. and the PDP research group (1986) *Parallel distributed processing Explorations in the microstructure of cognition.* Cambridge, MA: Bradford Books.

Salton, G., & McGill, M. J. (1983). *Introduction to modern infromation retrieval.* New York: McGraw-Hill.

Stritter, E. P. (1977). *File migration.* Doctoral dissertation, Stanford University, Stanford: Stanford Linear Accelerator Center.

Transitions in Strategy Choices

Robert S. Siegler

Carnegie-Mellon University

Traditionally, transitions in children's thinking have been characterized in elegantly simple ways. For example, 5-year-olds have been said to understand conservation in terms of a single dimension, the height of the liquid columns in the two glasses, whereas 7-year-olds have been said to understand it in terms of transformations. Similarly, first and second graders have been said to add small numbers by counting up from the larger addend, whereas those beyond third grade have been said to solve such problems by retrieving answers from memory (e. g., Groen & Parkman, 1972).

These models of transitions are clean and appealing, but they also are too simple to capture many of the most interesting changes in children's thinking. On a variety of problems, individual children know and use multiple strategies, not just one. They choose among these strategies in ways that produce adaptive combinations of strategy use at any one time and that produce adaptive changes in strategy use over time. They also construct new strategies, which gain a niche among the existing ones, and change the situations in which different strategies are used. Concurrent with all of these changes in strategy use, children become faster and more accurate in executing each strategy and in solving all problems.

This article is organized into four sections. The first describes some of the evidence that children actually use multiple strategies in situations where they previously have been presumed to just use one. The second describes a model of how children choose strategies at any particular point in development. The third describes the part of the model that deals with how children change strategies over time. The fourth describes how children construct new strategies.

Evidence that Children Use Diverse Strategies

Here and in most of the sections that follow, I describe research on 4- to 10-year-olds' strategy choices in arithmetic. My colleagues and I have obtained similar findings in a number of other areas: word identification, time-telling, spelling, and serial recall (Siegler, 1986; in press-a). I focus on the arithmetic research here because the findings are representative of those we have obtained in other domains, and because the models of transition processes have progressed the furthest.

For the past 15 years, the min model has been widely accepted as accurately describing the way that first and second graders solve addition problems. In this model, children consistently solve addition problems by counting up from the larger addend the number of times indicated by the smaller addend. For example, on 3+6, they would think "6, 7, 8, 9". The min model predicts that solution times on each problem will be a linear function of the smaller addend, because the smaller addend indicates the amount of counting-on from the larger number that needs to be done to solve the problem.

This prediction has proved accurate for both groups of children and individuals, in both Europe and North America, and in both standard and special education settings.

Despite all this support, the min model is wrong. Siegler (1987) examined young children's simple addition, using both the usual solution-time measures and children's verbal reports. The results were striking. When data were averaged over all trials (and over all strategies), as in earlier studies, the results replicated the previous finding that solution times were a linear function of the smaller addend. If these analyses had been the only ones conducted, the usual conclusion would have been reached, namely that first and second graders consistently use the min strategy to add.

However, the children's verbal reports suggested a quite different picture. The min strategy was but one of five approaches that they reported using. This reporting of diverse strategies characterized individual as well as group performance; most children reported using at least three approaches. Children reported using the min strategy on only 36% of trials.

Dividing the solution time data according to what strategy children said they had used on that trial lent considerable credence to the children's verbal reports. On trials where they reported using the min strategy, the min model was an even better predictor of solution times than in past studies or in the Siegler (1987) data set as a whole; it accounted for 86% of the variance in solution times. In contrast, on trials where they reported using one of the other strategies, the min model was never a good predictor of performance, either in absolute terms or relative to other predictors. It never accounted for as much as 40% of the variance. A variety of measures converged on the conclusion that children used the five strategies that they reported using, and that they employed them on those trials where they said they had.

Models of transitions can be no better than the characterizations of early and later knowledge states that they are attempting to connect. In arithmetic and many other domains, adequately characterizing these knowledge states demands recognition of children's use of diverse strategies over extended periods of development.

A Model of How Children Choose Which Strategy to Use

Once we recognize that children use diverse strategies to solve many problems, it becomes essential to identify how they choose among them. For the past few years, my colleagues and I have been developing a model of how children choose among their diverse strategies. The model has been implemented in detail (as a running computer simulation) for addition, subtraction, and multiplication. In all of these areas, the simulations produce strategy choices at any given time, changes in strategy use over time, and improvements in accuracy and speed much like those of the children we have observed.

The current version of the simulation, which I will describe here, is a more general version of the addition simulation described by Siegler and

Shrager (1984). Like the previous version, it includes a representation of knowledge and a process that operates on that representation to produce performance and learning.

First consider the representation. Children are hypothesized to have knowledge of problems, of strategies, and of the interaction between problems and strategies. Their knowledge of problems involves associations between each problem and possible answers to that problem, both correct and incorrect. For example, 5+3 would be associated not only with 8 but also with 6, 7, and 9. These representations of knowledge of each problem can be classified along a dimension of the peakedness of their distribution of associations. In a peaked distribution, most associative strength is concentrated in the correct answer. At the other extreme, in a flat distribution, associative strength is dispersed among several answers, with none of them forming a strong peak. For example, in Figure 1, the associative strengths for answers to 2+1 form a peaked distribution (with the associative strength for 3 at the peak) and those for 3+5 form a flat distribution.

The representation also includes knowledge about strategies. Each time a strategy is used, the simulation gains information about its speed and accuracy. This information generates a strength for each strategy, both in general and on particular problems. The strategies modeled in the current version of the addition simulation are the three most common approaches that young children use: counting from one, the min strategy, and retrieval.

One further feature of the representation should be mentioned. Newly generated strategies possess "novelty points" that temporarily add to their strength and thus allow them to be tried even when they have little or no track record. The strength conferred by these novelty points is gradually lost as experience with the strategy provides an increasingly informative data base about it. This feature was motivated by the view that people are often interested in exercising newly developed cognitive capabilities (Piaget, 1952), and by the realization that without a track record, a newly-developed strategy would be unlikely to be chosen.

Now consider the process that operates on this representation to produce performance. First, the process chooses a strategy. The probability of a

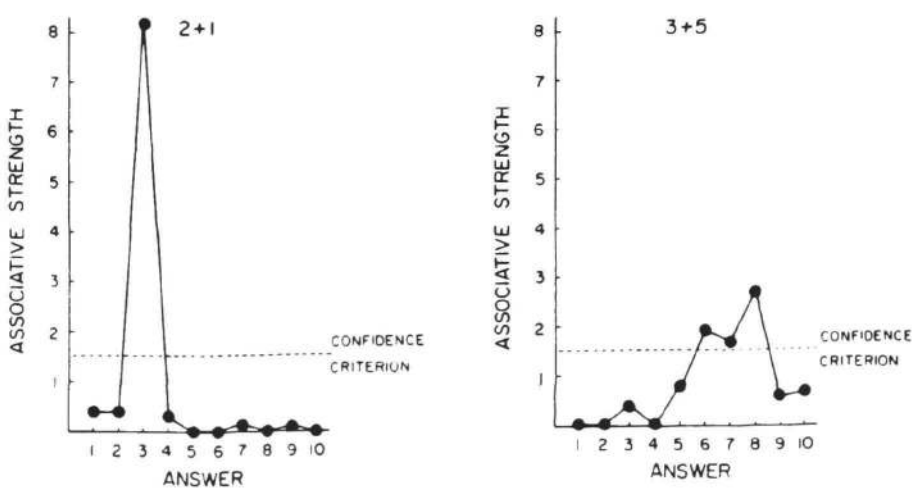


Figure 1. A peaked (left) and a flat (right) distribution of associations.

given strategy being chosen is proportional to its strength relative to the strength of all strategies. Strength of a strategy on a problem is a joint function of the local value of the strategy (how well it has done on that problem in the past) and of its global value (how well it has done across all problems). On problems never previously encountered, the global value of the strategy is the sole determinant of its strength. Thus, the stronger a strategy in general and on the particular problem that is posed, the more likely that it will be chosen for use on that problem.

If a strategy other than retrieval is chosen, that strategy is executed. If retrieval is chosen, the simulation retrieves a specific answer (e. g., 4) from the problem's distribution of associations (Figure 1). The probability of any given answer being retrieved is proportional to that answer's associative strength relative to the strength of all answers to the problem. Thus, in Figure 1, the connection between "2+1" and "3" has a strength of .80, the strength of connections between "2+1" and all answers is 1.00, so the probability of retrieving "3" is 80%. If the associative strength of whichever answer is chosen exceeds the confidence criterion (a threshold for stating a retrieved answer), the simulation states that answer. Otherwise, the simulation again chooses a strategy with probability proportional to the strength of that strategy relative to those of all strategies. The process continues until a strategy is chosen and an answer stated.

The simulation generates patterns of accuracy, solution times, and strategy use much like those of children. For example, it uses the min strategy most often on problems where the smaller of the two addends is very small and where the difference between the two addends is quite large. Siegler (1987) found the same pattern in kindergarteners', first graders', and second graders' performance. Also as with children, the simulation uses retrieval most often on problems where both addends are small and uses counting-from-one primarily on problems where both addends are large. Relative problem difficulty and particular errors that the simulation makes also parallel those of children. The reason lies in the simulation's learning mechanism, which is described in the next section.

Transitions in Strategy Use Over Time

The simulation learns a great deal through its experience with strategies and problems. As it gains experience, it produces faster and more accurate performance, more frequent use of retrieval, less frequent use of counting from one, and closer fitting of when strategies are used to their advantages and disadvantages on each problem. Such learning is not produced by any explicit, metacognitive governmental process, but rather through the operation of the above-described program together with a simple learning mechanism: children associate answers that they state with the problem on which they state them, and associate each strategy with the speed and accuracy that the strategy has produced on each problem and over all problems.

The way that this learning mechanism operates can be illustrated in the context of why some strategies are assigned to some problems more than others. Consider two problems, 9+1 and 5+5. Kindergarteners and first graders use the min strategy considerably more often on 9+1, yet use counting-from-one more

often on 5+5. The simulation generates similar behavior, and illustrates how such a pattern might emerge. On 9+1, the min strategy has a very large advantage in both speed and accuracy over the count-all strategy. It requires only 1/10 as many counts. In contrast, the numbers of counts required to execute the two strategies are more comparable on 5+5, where the min strategy requires 1/2 as many counts. If the number of counts were the only consideration, children might be expected to consistently use the min strategy on both problems (and on all problems) from the time they learned it. However, for any given number of counts, counting-on from an arbitrary number is considerably more difficult for young children (in terms of time and errors per count) than counting from one Fuson and Richards, 1982). The simulation's probability of erring on each count, and its time per count, reflect this greater difficulty of counting-on from a number larger than one. Thus, the simulation learns that although the min strategy is generally more effective, there are some problems, such as 5+5, where counting-from-one works better. This leads to counting-from-one being the most frequent strategy on such problems for awhile. It eventually is overtaken by retrieval, however, as the associative strength of the correct answer becomes sufficiently great that it is likely to be retrieved and stated.

The influence of performance on learning also is reflected in relative problem difficulty and in the particular errors that children make. Early in learning, children most often use backup strategies (such as counting-from-one and the min strategy). Early patterns of difficulty in executing such backup strategies seem to influence later patterns of retrieval difficulty and particular errors that are made. For example, in multiplication, the most common backup strategy is repeated addition. Repeated addition generates two main types of errors: answers in which one multiplicand is added too many or too few times (e. g., on 8X4, adding 7 or 9 4's, and getting 28 or 36) and small addition errors (e. g., adding 8 4's and getting 33). These are the same types of errors that children make most often when retrieving answers and that adults make under time pressure. Similarly, third and fourth graders' probability of correctly executing repeated addition on simple multiplication problems is highly correlated with their probability of retrieving correctly (Siegler, in press-a). The same relation between difficulty of solving problems via backup strategies and via retrieval has been found in addition and subtraction (Siegler, 1986).

The simulation's learning mechanism also produces parallels between backup strategy and retrieval performance. Problems that are more difficult to solve early in learning via backup strategies become more difficult to solve later via retrieval. When the simulation errs in using backup strategies, the result is less opportunity to associate the problem with the correct answer and more opportunity to associate incorrect answers with it.

Backup strategies also influence their own future use, in a somewhat ironic way. The more accurately that they are executed within the simulation, the sooner they stop being used. The reason is that accurate execution of backup strategies leads to peaked distributions of associations between answers and each problem, which in turn leads to retrieval, inherently the fastest strategy, also generating accurate performance and therefore being used increasingly often. Thus, backup strategies contribute to the transition

process in ways that lead to their own demise. Consistent with this view, the accuracy with which children execute backup strategies is quite strongly correlated with how often they are able to correctly retrieve answers in addition, subtraction, and word identification (Siegler, in press-b).

Acquisition of New Strategies

The computer simulations generate a fair range of transitions in children's arithmetic, but by no means all of them. Perhaps the most conspicuous gap is in the account of how new strategies are acquired. To learn more about this issue, Eric Jenkins, a graduate student at CMU, and I are currently conducting a longitudinal study of acquisition of the min strategy. Past studies indicate that children ordinarily acquire this strategy at age 5 or 6, and that 4-year-olds can learn it if given extensive addition experience.

To study the acquisition process, Jenkins and I pretested a group of 4- and 5-year-olds to identify children who gave no evidence of prior knowledge of the min strategy. Once the children were identified, we presented them 7 problems daily for about 30 sessions. After each problem, we asked the child how he or she had solved the problem. The verbal statement, together with the videocassette of the child's overt behavior while solving the problem, became our guide for identifying the child's strategy on each trial. This gave us a way of identifying the exact trial on which the child discovered the min strategy, and thus to analyze what led up to the discovery and how the strategy, once discovered, was extended to new problems.

The min strategy was discovered by 7 of the 9 children in the experiment. Some children constructed the strategy within the first 5 sessions; others did so between the 25th and 30th sessions. For some children, invention of the new strategy was accompanied by conscious appreciation that they were doing something new and that the new way of adding was more efficient. Other children were unaware that they were doing anything different, even saying that they had counted from one when they had audibly began counting at the larger addend.

The most striking finding of the study involved a condition that seemed to strongly promote both discovery of the min strategy among children who had not already discovered it and increased use of the strategy among those who had. This condition involved presentation of problems that would be very difficult to solve by means of the prior counting-from-one strategy, but that would be quite easy to solve via the min strategy. These problems involved adding very large and very small numbers, scuh as 24+3 and 2+23. Within two sessions of the introduction of these problems, use of the min strategy increased from 15% to 65% of trials on which any type of counting was used. The progress was maintained after the demanding problems were no longer given and more standard problems (e. g., 7+4) were substituted. Thus, children met the challenge of the difficult problems by constructing a new strategy, and then continued to use the new strategy on other problems.

Looking at the present research as a whole, perhaps the most striking characteristic of cognitive-developmental transitions is their self-regulating

quality. The transitions are self-regulating in at least three senses. First, children's choices of strategies adapt to changing circumstances. As they gain experience with a strategy, they use it increasingly often on those problems where its advantages relative to other strategies are greatest. Second, children's strategy choices have built into them a kind of self-righting capability, an ability to recover from errors and initial unfavorable experiences. The heavy use of backup strategies early in learning confers this type of stability. Illustratively, the simulation of multiplication reported in Siegler (in press-a) erred on its first 4 answers on 8X9 and on 8 of its first 10. Yet by the end of the learning phase, the simulation was advancing the correct answer on 99% of trials. The reason was that over trials, the backup strategies produced 72 more often than any other answer, which led to its associative strength increasing and therefore to its being retrieved and stated increasingly often.

Third, when existing procedures prove inadequate, children are especially likely to create new strategies that can overcome the difficulties. The present models produce the first two types of self-regulation; I hope soon to incorporate mechanisms into the models, perhaps akin to those in Newell's Soar or VanLehn's Sierra models, that produce the third type of self-regulation as well.

References

Fuson, K. C. & Richards, J. (1982). The acquisition and elaboration of the number sequence. In C. Brainerd (Ed.) Progress in Cognitive Developmental Research: Vol. 1. New York: Springer-Verlag, 1982.

Groen, G. J., & Parkman, J. M. (1972) A chronometric analysis of simple addition. Psychological Review, 79, 329-343.

Piaget, J. (1952). The origins of intelligence in children. New York: International Universities Press.

Siegler, R. S. (1986). Unities across domains in children's strategy choices. In M. Perlmutter (Ed.) Minnesota Symposium on Child Psychology, Vol. 19, (pp. 1-48), Hillsdale, NJ: Erlbaum.

Siegler, R. S. (1987). The perils of averaging data over strategies: An example from children's addition. Journal of Experimental Psychology: General, 116, 250-264.

Siegler, R. S. (In press-a). Strategy choice procedures and the development of multiplication skill. Journal of Experimental Psychology: General.

Siegler, R. S. (In press-b). Individual differences in strategy choices: Good students, not-so-good students, and perfectionists. Child Development.

Siegler, R. S., & Shrager, J. (1984). A model of strategy choice. In C. Sophian (Ed.), Origins of cognitive skills (pp. 229-293). Hillsdale, NJ: Erlbaum.

VITAL

A Connectionist Parser

Tim Howells

Department of Computer and Information Science
University of Massachusetts at Amherst

Abstract

VITAL is an experiment in parsing natural language through the interaction of many local processes without an explicit global control. The global interpretation is achieved through the convergence of a network of processes on a mutually consistent state through cycles of spreading activation and an implicit form of mutual inhibition. This can be viewed as a constraint satisfaction technique consistent with current research both in linguistics and connectionism.[1]

Keywords: Natural language processing, Connectionism

Natural language as constraint satisfaction

In linguistics, computational linguistics and natural language processing the idea is becoming increasingly prevalent that language is better described as the interaction of many weak constraints than as the interpretation of a rigid rule system (Berwick, 1987). Government Binding Theory, Lexical Functional Grammar and Generalized Phrase Structure Grammar all seem to favor a cooperative control structure governed by a multitude of very specific, local constraints distributed throughout the lexicon. Similarly, the Word Expert Parser (Small & Rieger, 1982) has attracted much interest in natural language processing circles as an attempt to define a real formalism based on the sorts of complex lexical interactions that are implicit in the Yale style conceptual analyzers (Birnbaum & Selfridge, 1981). The appeal of this approach is its promise of a more flexible mechanism, with the ability to handle "more or less" grammatical utterances, and to provide reasonable performance in the presence of flaws and shortcomings here and there in the system. It also provides a processing model more consistent with what we know of cognitive processing, especially if the local constraints can be treated as fairly autonomous parallel processes. A problem with the approach is obvious: given a multitude of interacting agents, each with its own local view, how do you control the flow of processing to arrive at a global interpretation at all?

Connectionist models

Connectionist network relaxation models provide an attractive approach to this sort of constraint satisfaction problem (Smolenski, 1986). Communication between processes is limited

[1]This research was supported by an NSF Presidential Young Investigators Award NSFIST-8351863, DARPA contract N00014-87-K-0238, and the Office of Naval Research under a University Research Initiative grant, contract N00014-86-K-0764.

to the passing of numerical activation and inhibition, with the processing state at any point in time defined by the distribution of activation. If all goes well the network will converge on a configuration that represents a maximally consistent global interpretation. This is a truly distributed control structure with a very high degree of parallelism. It also requires that the system be developed according to simple and consistent principles, which may be a benefit in itself.

Gary Cottrell and Steven Small have used this approach to model word sense disambiguation (Cottrell & Small, 1983). Jordan Pollack and David Waltz have proposed it as a way of integrating multiple knowledge sources in natural language processing (Waltz & Pollack, 1985). The systems most similar to my own are the parsers built by Mark Fanty (Fanty, 1985) and Bart Selman (Selman & Hirst, 1985). These both parse according to a phrase structure grammar using spreading activation over a network in which the nodes represent the rules in the grammar. The network converges with a parse tree of the input sentence activated.

VITAL

Distinguishing characteristics

While Mark Fanty used a *dynamic programming* approach and Bart Selman a *Boltzmann machine* approach in their parsers, I've applied an interactive relaxation algorithm similar to that used in McClelland and Rumelhart's word recognition program (1981). I feel that this technique maps more naturally onto the problem, although the network behavior is not as well understood as it is with either of the other two approaches. While Selman and Fanty used prebuilt data structures capable of representing all possible parses of all sentences of up to a fixed length, my networks are built dynamically during parsing. I regard this as a question of implementation with the usual sorts of space/time trade offs.

My relaxation algorithm is unusual in that instead of explicit inhibitory links it utilizes decay over time together with a "competition" for available activation similar to that proposed by James Reggia (1987). This produced a number of nice results which will be detailed below.

The background grammar

The phrase structure grammar is encoded as a collection of "template" network nodes, each of which represents a single rule. The essential information in these template nodes is:

- The rule itself; *e.g.* **NP** $\Rightarrow$ **DET** $\overline{\mathbf{N}}$ [2]
- Pointers to other rules (I'll call them *context rules*) that have the represented rule's left-hand-side as a constituent of their right-hand-sides. Two context rules for the above rule would be:
  1. **S** $\Rightarrow$ **NP VP**
  2. **PrepPhr** $\Rightarrow$ **Prep NP**
- Weights reflecting the statistical likelihood that the represented rule is in fact a constituent of each of those context rules, assuming that the represented rule is in the parse tree. These are simply counters that are incremented every time the two rules participate together in a successful parse.

[2] $\overline{\mathrm{N}}$ (N bar) is an intermediate stage in forming a noun phrase that may be completed by tacking on a determiner. It may be a single noun or a complex phrase like "woman that we saw on the way to the park".

Table 1: A sample grammar fragment

syntactic unit	*context rules*	*count*
book	N ⇒ book	18
	V ⇒ book	3
DET	NP ⇒ DET $\overline{\text{N}}$	208
N	$\overline{\text{N}}$ ⇒ N	172
$\overline{\text{N}}$	NP ⇒ DET $\overline{\text{N}}$	208
	$\overline{\text{N}}$ ⇒ ADJ $\overline{\text{N}}$	35
the	DET ⇒ the	72

As needed during parsing, copies of these "templates" are instantiated as nodes in the network.

A distinction is made between "branching" rules that have more than one constituent on their right hand sides, and "non-branching" rules which have only a single constituent. There is a qualitative difference between these two classes of rules: a branching rule specifies a way in which its constituents may be combined, while a non-branching rule merely provides another name by which to refer to its constituent.

A portion of the grammar needed for the parsing of noun phrases might be recorded as shown in the above table. In this example, the parser has currently seen a total of 243 instances where an $\overline{\text{N}}$ node was incorporated into a parse tree. In 208 of those cases it was simply joined with a determiner to form a noun phrase (NP) as in "the house". In the remaining 35 cases where an $\overline{\text{N}}$ node occurred it was combined with an adjective to form a new $\overline{\text{N}}$ such as "red house". The rules shown here are oversimplified for the sake of an easy example.

A general description of the algorithm

Network building

Network building begins with the instantiation of the words of the input sentence as the terminal nodes. These are given an activation of 200 which remains constant throughout the parse. When a node comes into existence it first activates all its non-branching context rules. These are activated immediately because they merely represent alternative ways to refer to the original node and not a further hypothesis regarding how other consitituents might be combined. As nodes in the network reach a threshold activation level (currently set at 110) they activate new nodes corresponding to their branching context rules. The new nodes are initialized with an activation level proportional to the activation levels of their constituents and the counters associated with its context rule. I'll give an example of this below. When a node reaches an activation level of zero it is removed from the network.

The process of network building is illustrated in *figure 1*, which shows three steps in parsing the noun phrase "the book" using the grammar of *table 1*.

(a) The input nodes are instantiated along with their non-branching context-nodes (and their non-branching context nodes' non-branching context-nodes, *etc.*).

(b) As existing nodes pass the threshold activation level they activate new nodes representing their context-nodes. These new nodes look on lists of existing nodes to find their other constituents. Internal bookeeping prevents the creation of duplicate nodes.

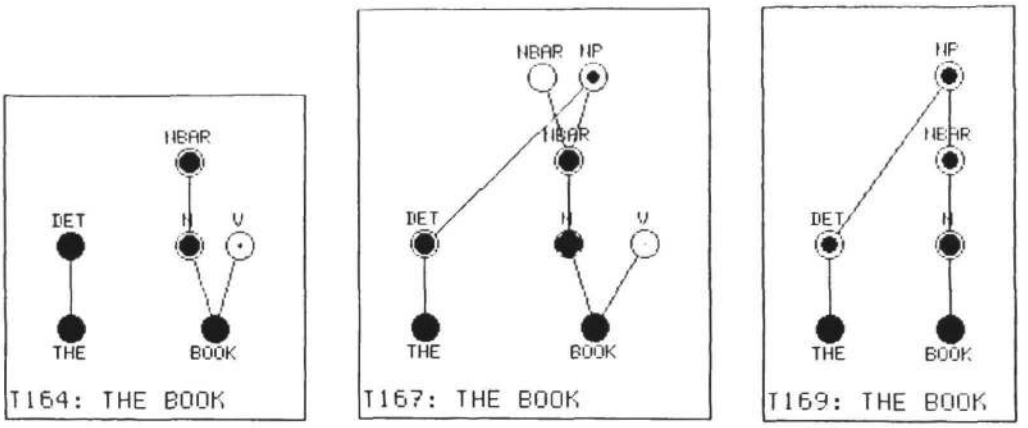


Figure 1: parsing "the book"

(c) During the network relaxation superfluous nodes reach zero activation levels and are removed from the network.

To give an example of how activation levels are initialized, when the "book" node activated its context rules "N" and "V" it had an activation level of 200; the frequency counts it had were 208 for "N" and 35 for "V" (totaling 243); therefore it created an "N" hypothesis with an activation of $200 \times \frac{208}{243}$ or 171, and a "V" hypothesis with an activation of $200 \times \frac{35}{243}$ or 29. As nodes aquire new constituents they immediately get more activation from them according to the same formula. The effect of this procedure with realistic grammars is that syntactic units like $\overline{\mathrm{N}}$ that participate in many different constructions will activate many relatively weak hypothesis, while the function words (*e.g.* articles and prepostitions) will activate only a small number of very strong hypotheses.

Network relaxation

With each cycle each node sends some portion of its total activation to its connected nodes and then the new activation levels are subject to decay. The terminal nodes representing the input send activation but do not receive it and are not subject to decay. There are distinct procedures for sending activation to the nodes that want the sending node as a constituent (bottom-up activation) and the nodes that *it* wants as constituents (top-down activation):

- *Bottom-up activation:* The total amount of activation to be divided among the nodes competing for the sending node as a constituent is calculated as a percentage of its activation. Then it's apportioned according to the square of their current activation levels, so that nodes that already have high activation levels attract large amounts of activation from their constituents, while those with low activation attract little. Say the system parameter controling bottom-up activation is P (I'm currently setting P at 0.4). If a node with an activation level of A is sending bottom-up activation to n other nodes and the activation of the ith of those n nodes is a_i. Then the activation given to the ith node will be:

$$A \times P \times \frac{a_i^2}{\sum_{j=1}^{n} a_j^2}$$

- *Top-down activation:* An amount of activation, again equal to a percentage (currently 40%) of the sending node's activation, is divided equally among the sending node's constituents.

These procedures reflect the fact that each node in the final parse should have *all* its available constituent nodes, but should only be a constituent of a single node, hence the winner take all situation for bottom-up activation.

After each cycle of bottom-up and top-down activation the nodes are subject to a decay rate controled by another system parameter (currently set at 0.8). A penalty for nodes that do not have all of their consituents is factored in. If the parameter is P, a node's activation before decay is A, the number of constituents it requires is C_r and the number of constituents it actually has is C_a then after decay its activation is:

$$A \times (1 - P) \times \frac{C_a}{C_r}$$

An example

Figures two through six show the network at various stages while parsing the sentence "Mary gave the woman wearing a yellow dress the book." The parse is complicated by the fact that the word "yellow" is not in the lexicon. *VITAL* uses top-down expectations to complete the pattern. The dotted lines in the network graphs indicate the "guesses" it made. In this example the parse tree was derived in 24 cycles and the network converged in 34 cycles.

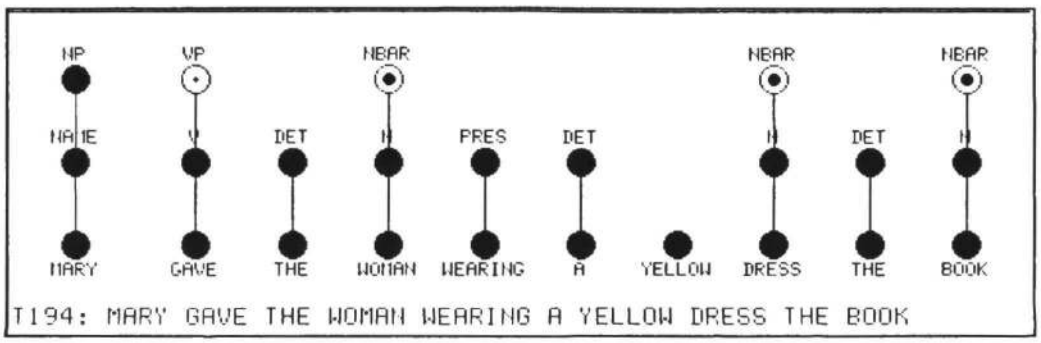


Figure 2: Instantiation of the input nodes

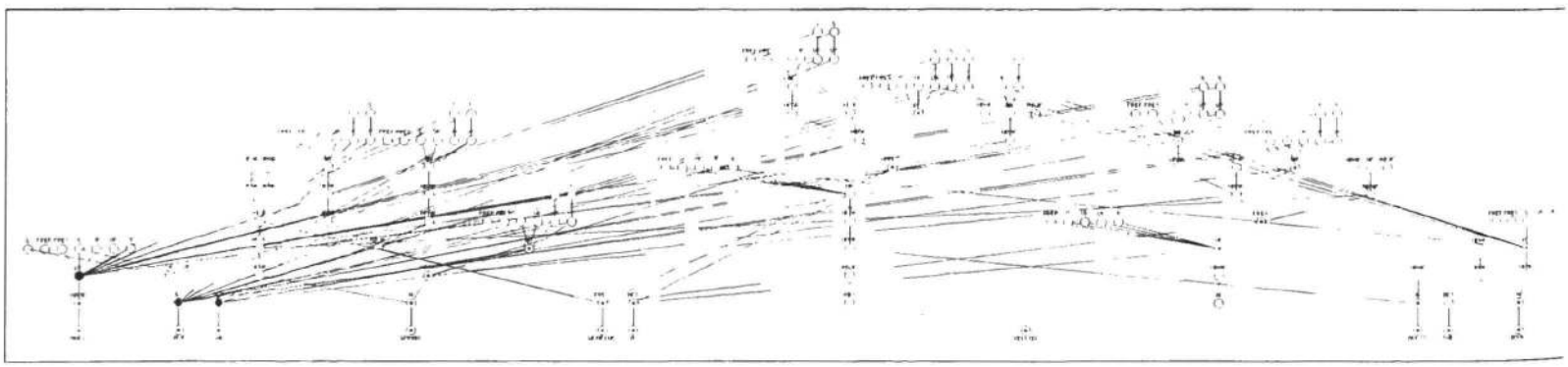


Figure 3: The network after 1 cycle

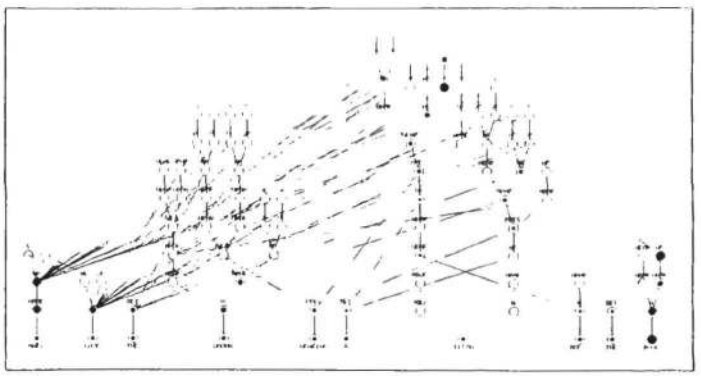


Figure 4: ...after 5 cycles

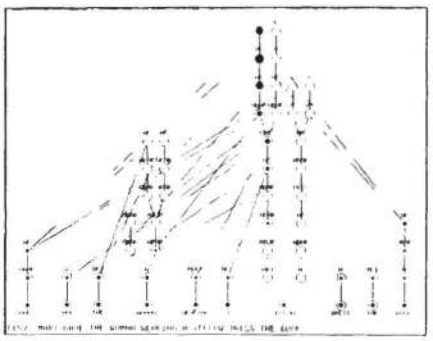


Figure 5: ...15 cycles

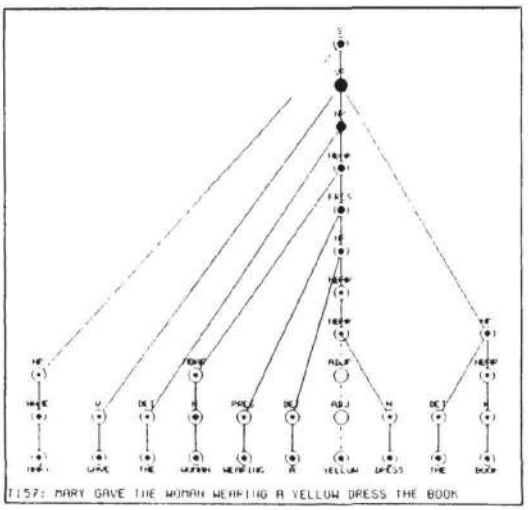


Figure 6: The network after 25 cycles

Discussion

A nice feature of this algorithm is a smooth integration of bottom-up and top-down processing. In the early cycles the activation is strongest in the nodes at the bottom of the network. After several cycles the activation is distributed more evenly, and in the late cycles the heaviest concentration of activation is in a few nodes near the top of the network. The more usual technique of using explicit inhibitory links requires the use of a squashing function to keep the activation levels within preset bounds. By using dynamically changing weights and decay, the nodes converge on values reflecting the amount of activation available to them across the network. Island driving effects arise naturally, since nodes representing hypotheses with a complex and unambiguous internal structure attract large amounts of activation relative to simpler or confused hypotheses. The use of inhibitory links also entails a much more highly interconnected network (Reggia, 1987).

Hinton and Sejnowski have described a major concern regarding relaxation algorithms:

> We would like these networks to settle into states in which a few units are fully active and the rest are inactive. Such states constitute clean "digital" interpretations. To prevent the network from hedging its bets by settling into a state where many units are slightly active, it is usually necessary to use a strongly nonlinear decision rule, and this also speeds up convergence. (Hinton and Sejnowski, 1986)

With my algorithm I found that I could get the desired "clean interpretations" with real-valued units by allowing a very active flow of energy through the system. That is to say the network would settle with all nodes either highly activated or completely off if I set the system parameters so that the nodes send large amounts of activation and are subject to a high rate of decay. [3]

A good test of this behavior is highly ambiguous input where you might expect the activation to get distributed among the equally valid interpretations rather than settling decisively on one of them. I tested this with the noun phrase "the green green dress wearing woman" which has three equally valid syntactic interpretations. *VITAL* generates all three interpretations in its network but settles cleanly on a single one. This is a particularly important quality in natural language processing, since a typical sentence in a business letter can have a few hundred syntactic interpretations. It would be nice to be able to bias the network in favor one or another interpretation, but in order to do this effectively I'll have to introduce some additional sophistication in the network nodes.

I think that this network relaxation algorithm is an intuitively pleasing style of parsing. As described above, a simple distributed control structure allows bottom-up and top-down effects to be integrated smoothly and effectively, and the procedure for initializing the activation levels of new nodes based on the local statistical memory of the existing nodes causes function words to dominate processing in a natural way. This memory based approach also leaves room for learning over time, although this is not a learning program in the usual connectionist sense. The next natural step with this parser would be to incorporate a semantic component that does limited inference using the same constraint satisfaction approach with local associative memories. There are formidable problems in trying to extend this approach to semantic processing, however, mainly because we can't represent semantic structures as cleanly as the syntactic structures of a context-free grammar.

Acknowledgements

Thanks to Wendy Lehnert for her support and for introducing me to these ideas with her *ELAN* program. Also to Steve Bradtke for many useful discussions and criticisms, and for his graphics package, without which this might never have gotten off the ground.

References

Birnbaum, L. and M. Selfridge. 1981. "Conceptual analysis of natural language". in *Inside Computer Understanding*. ed. Schank and Riesbeck. Hillsdale, NJ: Lawrence Erlbaum

Berwick, R.C.. 1987. "Principle based parsing". technical report AI-TR-972, Artificial Intelligence Laboratory, The Massachusetts Institute of Technology, Cambridge, MA. to appear in *The Processing of Linguistic Structure*. Cambridge: MIT Press

Cottrell, G.W. and S.L. Small. 1983. "A Connectionist Scheme for Modelling Word-Sense Disambiguation". *Cognition and Brain Theory* 6, 89-120

Fanty, M.. 1985. "Context-free parsing in connectionist networks". TR-174. Computer Science Department, the University of Rochester, Rochester, NY 14627

[3] I currently am having the nodes send out 80% of their activation levels to their connected nodes and am subjecting them to an 80% decay rate with each cycle.

Hinton, G.E. and T.J. Sejnowski. 1986. "Learning and relearning in Boltzmann Machines". in Rumelhart, McClelland *et al.*, 1986

Lehnert, W.G.. 1987. "Learning to integrate syntax and semantics". *Proceedings of the Fourth International Conference on Machine Learning.* Irvine, CA

McClelland, J.L. and D.E. Rumelhart. 1981. "An interactive activation model of context effects in letter perception: Part 1. An account of basic findings". *Psychological Review*, 88, 375-407

Reggia, J.A.. 1987. "Properties of a competition-based activation mechanism in neuromimetic network models". *Proceedings of the IEEE First International Conference on Neural Networks*, II,131.. San Diego, CA

Rumelhart, D.E., J.L. McClelland *et al.*. 1986. *Parallel Distributed Processing; vol. 1.* Cambridge: MIT Press

Selman, B. and G. Hirst. 1985. "A rule based connectionist parsing system". *Proceedings of the Seventh Annual Conference of the Cognitive Science Society.* Irvine, CA, 212-221

Small, S. and C. Rieger. 1982. "Parsing and comprehending with word experts (a theory and its realization)". in *Strategies for Natural Language Processing*, ed. Lehnert and Ringle. Hillsdale, NJ: Lawrence Erlbaum

Smolenski, Paul. 1986. "Information processing in dynamical systems: Foundations of harmony theory". in Rumelhart, McClelland *et al.*, 1986

Waltz, D.L. and J.B. Pollack. 1985. "Massively parallel parsing". *Cognitive Science*, 9, 51-74

Applying Contextual Constraints in Sentence Comprehension

Mark F. St. John and James L. McClelland

Carnegie-Mellon University

The goal of our research has been to develop a model that converts a simple sentence into a conceptual representation of the event that the sentence describes: specifically, a model that converts the constituent phrases of a sentence into a representation of an event, that assigns a thematic role to each constituent (Fillmore, 1968), and that interprets ambiguous and vague words. In our model, the comprehension process is viewed as a form of constraint satisfaction. The surface features of a sentence, its particular words and their order and morphology provide a rich set of constraints on the sentence's meaning. These constraints vary in strength and compete or cooperate according to their strength to produce an interpretation of the sentence.

Determining the exact constraints, and their appropriate strengths, is difficult, but a connectionist learning procedure allows a model to learn them. The learning procedures take a statistical approach to the task. By comparing large numbers of sentences to the events they describe, the many-to-many mapping, between features of the sentences and events, emerges as statistical regularities.

Often a sentence will omit information about the event it describes. We wanted our model to infer this missing information: to represent the event as completely as possible. Sometimes, though, a sentence is compatible with more than one interpretation. In "The private shot the target," the instrument is sometimes a rifle, and sometimes a pistol. In such situations, a good long-term strategy is to represent each possibility according to its likelihood in the given context.

As each constituent of a sentence is processed, the context changes. The processor should update its inferences to reflect the changing context. It should also adjust its interpretation of previous material. The model should utilize information derived from the sentence immediately (Carpenter & Just, 1977; Marslen-Wilson & Tyler, 1980). In all, therefore, we have six goals for our model of sentence comprehension:

* to disambiguate ambiguous words
* to instantiate vague words
* to assign thematic roles
* to elaborate implied roles
* to learn to perform these tasks
* to immediately adjust its interpretation

Overview of the Model

Processing

A sentence is represented as a sequence of phrases and each is processed in turn. The information each phrase provides is immediately used to update a representation of the event. This representation is called the sentence gestalt because all of the information from the sentence is represented together within a single, distributed representation. The event described by a sentence is represented as a pattern of activity across the units of this representation.

To process the constituent phrases of a sentence, we adapted an architecture from Jordan (1986) that uses the output of previous processing as input on the next iteration. To process a phrase, the appropriate *phrase* units are activated and the *sentence gestalt* activations (initially zero) are copied over to the *previous sentence gestalt* units. Activation from these layers combine in a hidden layer and create a new pattern over the *sentence gestalt* units (see Figure 1). Each phrase of the sentence is processed in sequence.

Though other models have used a type of sentence gestalt to represent the meaning of a sentence (McClelland & Kawamoto, 1986; St. John & McClelland, 1987), ours is the first to make the gestalt a trainable layer of hidden units. The advantage is that the network can optimize its representation to include only information that is relevant to its task. Since a layer of hidden units cannot be trained directly, we invented a way of "decoding" the representation into an output layer. The output layer represents the event as a set of thematic role and filler pairs. For example, the event described by "The pitcher

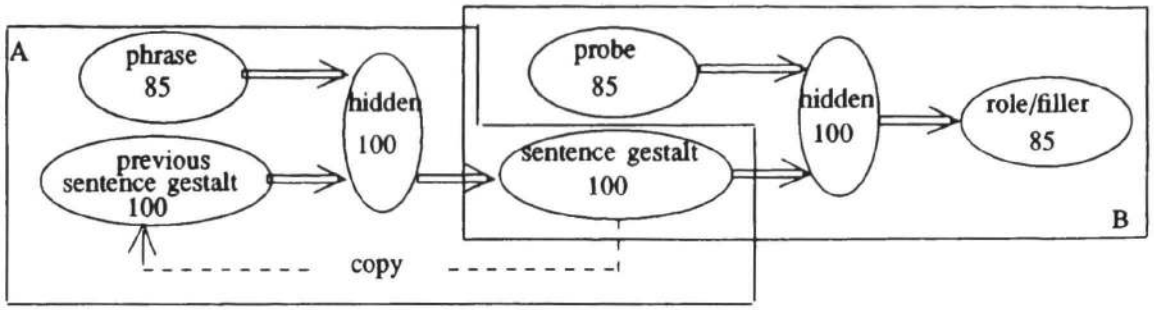


Figure 1. The architecture of the network. The boxes highlight the functional parts: Area A processes the phrases into the sentence gestalt, and Area B processes the sentence gestalt into the output representation.

threw the ball" would be represented as the set {(agent, pitcher/ball-player), (action, threw/toss), (patient, ball/sphere)}.

The output layer can represent one role/filler pair at a time. To decode a particular role/filler pair, the sentence gestalt is probed with half of the pair. Activation from the probe and the *sentence gestalt* combine in another hidden layer which then activates the entire pair in the output layer. The entire event can be decoded in this way by successively probing with each half of each pair.

When more than one object can plausibly fill a role, the model maximizes its long-term success by activating each filler to the degree it is likely. More formally, the activation of each filler should correspond to its conditional probability of occurring in the given context. The network should learn weights to produce these activations through training. To achieve this goal, we employed an error measure in the learning procedure, cross-entropy (Hinton, 1987), that converges on this goal:

$$C = - \sum_j [T_j \log_2 (A_j) + (1-T) \log_2 (1-A_j)]$$

where T_j is the target activation and A_j is the output activation of unit j. As with any connectionist learning procedure, the goal is to minimize the error measure or cost-function (cf. Hinton, 1987). When C is minimized across all the sentences of the training corpus, the activation of a particular output unit is equal to the conditional probability that whatever the unit represents is true given the current situation. Specifically, if each unit represented the occurrence of a particular filler in an event, that unit's activation would represent the conditional probability of that filler occurring given what was currently known about the event. In minimizing C, the network is searching for weights that allow it to match activations to conditional probabilities.

Environment and training

Training consists of trials in which the network is presented with a sentence and the event it describes. The network is trained to generate the event from the sentence as input. These sentence/event pairs were generated on-line for each training trial. Some pairs were more likely to be generated than others. Over training, these likelihood differences translated into differences in training frequency and created regularities.

To promote immediate processing a special training regime is used. After each phrase has been processed, the network is trained to predict the set of role/filler pairs of the entire sentence. From the first phrase of the sentence, then, the model is forced to try to predict the entire event. This training procedure forces the model to do as much immediate processing as possible. Consequently, as each new phrase is processed, the model's predictions of the event are refined to reflect the additional evidence it supplies.

An illustration of processing

As the network processes "The adult ate the steak," the *sentence gestalt* can be probed to see what it is representing after each phrase is processed (see Figure 2). After processing "the adult," the gestalt represents that the agent of the event is an adult, and it

guesses weakly at a number of actions. Following "ate," the network encodes that information and expects the patient to be food. Once "the steak" is processed, the network represents steak as the patient and infers that knife is the instrument. It also is able to revise its representation of the agent. Because one person, the busdriver, is the most frequent steak-eater in the corpus, the network infers that the adult from the sentence is the busdriver. In this way, the model infers missing thematic roles and immediately adjusts previous results (in this case by instantiating the agent) as more information becomes available.

Specifics of the Model

Input representation

Each phrase was encoded by a single unit representing the noun or verb. No semantic information was provided. For prepositional phrases, the preposition was encoded by a second unit. The passive voice was encoded by a unit in the verb phrase representation. One unit stood for each of 13 verbs, 31 nouns, 4 prepositions, 3 adverbs, and 7 ambiguous words. Two of the ambiguous words had two verb meanings, three had two noun meanings, and two had a verb and a noun meaning. Six of the words were vague terms (e.g. someone, something, and food).

The relative location of each phrase within the sentence was also encoded in the input. It was coded by four units that represent location respective to the verb: pre-verbal, verbal, first-post-verbal, and n-post-verbal. The first-post-verbal unit was active for the phrase immediately following the verb, and the n-post-verbal unit was active for any phrase

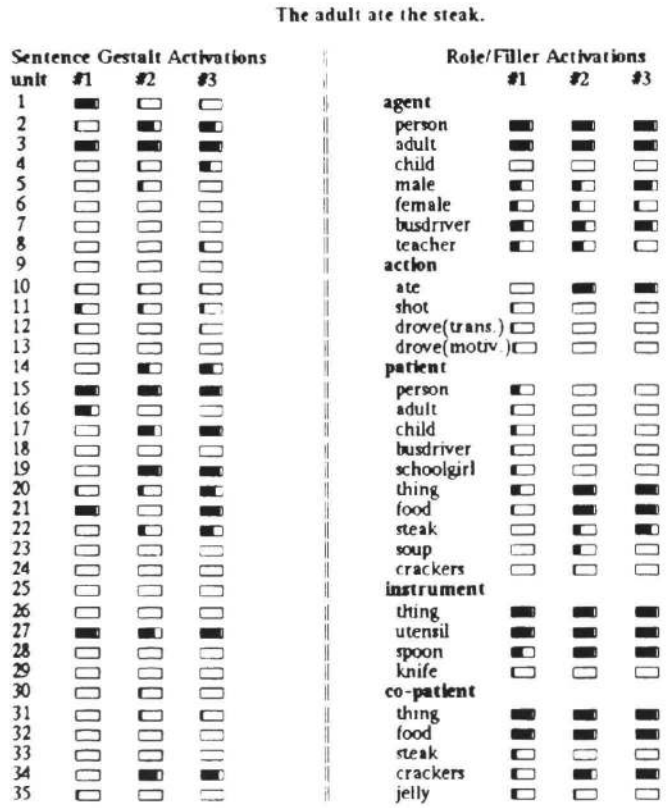


Figure 2. The evolution of the sentence gestalt during processing of a sentence. The # corresponds to the number of phrases that have been presented to the network at that point. #1 means the network has seen the first phrase; #2 means it has seen the first two phrases; etc. The activations (ranging between 0 and 1) of a sampling of units are graphed as the darkened area of each box. By probing the gestalt with the role half of each role/filler pair, the event represented by each gestalt can be observed. The role probe and the activation level of active filler units in the output layer are presented for each gestalt.

occurring after the first-post-verbal phrase. A number of phrases, therefore, could share the n-post-verbal position. For example, "The ball was hit by someone with the bat in the park," was encoded as the ordered set {(pre-verbal, ball), (verbal, passive-voice, hit), (first-post-verbal, by, someone), (n-post-verbal, with, bat), (n-post-verbal, in, park)}.

Output representation

The output had one unit for each of 9 possible thematic roles (e.g. agent, action, patient, instrument) and one unit for each of 28 concepts, 14 actions, and 3 adverbs. Additionally, 13 "feature" units, like male, female, and adult, were included in the output to allow the demonstration of more subtle effects of constraints on interpretation. Any one role/filler pattern, then, consisted of two parts: for the role, one of the 9 role units was active, and for the filler, a unit representing the concept, action, or adverb was active. If relevant, any of the feature units or the passive voice unit were active.[1] This representation is not meant to be comprehensive. Instead, it is meant to be a convenient way to train and demonstrate the processing abilities of the network.[2]

Training environment

While the sentences often include ambiguous or vague words, the events are always specific and complete: each thematic role related to an action is filled by some specific concept. Accordingly, each action occurs in a particular location, and actions requiring an instrument always have a specific instrument. The event would be generated by picking concepts to fill each thematic role related to the event according to preset probabilities. The probability of selecting each concept depended upon what else had been selected for that event. Conversely, the sentences often omit thematic roles (e.g. instrument and location) from mention and use vague words (e.g. someone and something) to refer to parts of an event.

The sentences had to be limited in complexity because of limitations in the event representation. Only one filler could be assigned to a role in a particular sentence. Also, all the roles are assumed to belong to the sentence as a whole. Therefore, no embedded clauses or phrases attached to single constituents were possible.

On each training trial, the error, in terms of cross-entropy, was propagated backward through the network (cf. Rumelhart, Hinton, & Williams, 1986). The weight changes from each trial were added together and used to update the weights every 60 trials. This consolidation of weight changes tends to smooth out the aberrations caused by the random generation of training examples.

Performance

Processing in general

The simulation was stopped and evaluated after 330,000 sentence trials. First, we will assess the model's ability to comprehend sentences generally. Then we will examine the model's ability to fulfill our specific processing goals. One hundred sentences were drawn randomly from the corpus. The probability of drawing a sentence was the same as during training. Consequently, frequently practiced sentences were more common among the 100 test sentences than infrequently practiced sentences. Thirteen of these sentences were completely ambiguous. For example, the sentence, "The adult drank the iced-tea," can be instantiated with either busdriver or teacher as the agent, but the sentence offers no clues that busdriver is the correct agent in this particular sentence/event pair in the test set.

---

[1] If the word related to the concept appeared in a prepositional phrase, such as "with the knife," the appropriate preposition unit was also activated.

[2] A second output layer was included in the simulations. This layer reproduced the input phrase that fit with the role/filler pair being probed. Consequently, the model was required to retain the specific words in the sentence as well as their meaning. Since this aspect of the processing does not fit into the context of the current discussion, these units are not discussed further.

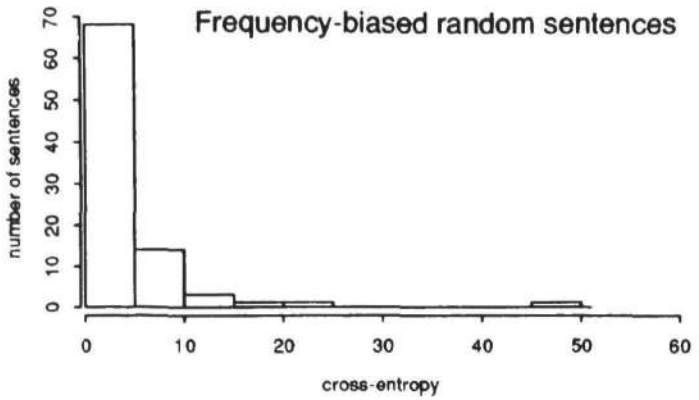


Figure 3. Histogram of the cross-entropy for frequency-biased random sentences. Sentences that were processed almost perfectly produced a small cross-entropy measure of between 0 and 10: only small errors occurred when an output unit should have been completely activate (with a value of 1), but only obtained an activation of .7 or .8, or when a unit should have had an activation of 0, but had an activation of .1 or .2. The incorrect activation of one role/filler pair produced a larger cross-entropy: between 15 and 20. For example, if teacher were supposed to be the agent, but the network activated busdriver, an error of about 15 would result.

Since the network cannot predict the event, these sentences were not tested. The remaining sentences were tested, and figure 3 presents a histogram of the results. Nearly every sentence was processed correctly.

Performance on specific processes

Our specific interest was to develop a processor that could correctly perform several important language comprehension processes. Five typical sentences were drawn from the corpus to test each processing task. The role assignment category was divided into groups according to the primary type of information used. Sentences in the active-voice semantic group contain semantic information relevant to assigning roles. In the example in Table 1, the network assigns schoolgirl to the role of agent. Of the concepts referred to in the sentence, only the schoolgirl has features which match an agent of stirring. Similarly, semantic knowledge constraints kool-aid to be the patient. Sentences in the passive-voice semantic category work similarly. The model processed each test sentence correctly.

To process sentences in the active and passive word order categories, however, the network cannot rely entirely on semantic constraints to assign thematic roles. In the sentence, "The busdriver was kissed by the teacher," the busdriver is as likely to be the

Processing tasks

Category	Example
Role assignment	
Active semantic	The schoolgirl stirred the kool-aid with a spoon.
Passive semantic	The ball was hit by the pitcher.
Active word order	The busdriver gave the rose to the teacher.
Passive word order	The busdriver was kissed by the teacher.
Word ambiguity	The pitcher hit the bat with the bat.
Concept instantiation	The teacher kissed someone.
Role elaboration	The teacher ate the soup (with a spoon).

Table 1. The four categories of processing tasks and an example sentence of each. Role assignment was tested under four conditions to assess the use of both semantic-and word order information. The remaining three categories involve inferences about the content of a sentence. The parentheses in the role elaboration example denote a role that is not presented in the sentence, and must be inferred from the context.

agent as the patient. Only the relative location information, in conjunction with the passive cues, can cue the correct role assignments. In active sentences, the pre-verbal phrase is the agent and the post-verbal phrase is the patient. In passive sentences, on the other hand, the pre-verbal phrase is the patient and the post-verbal phrase is the agent. The model processed each test sentence correctly.

The remaining three categories involve the use of context to further specify the concepts referred to in a sentence. Sentences in the word ambiguity category contain ambiguous words. While the word itself cues two different interpretations, the context fits only one. In "The pitcher hit the bat with the bat," pitcher cues both container and ball player. The context cues both ball player and busdriver because the model has seen sentences involving both people hitting bats. All the constraints supporting ball player combine, and together they win the competition for the interpretation of the sentence. Even when several words of a sentence are ambiguous, the event which they support in common dominates the disparate events that they support individually. Consequently, the final interpretation of each word fits together into a globally consistent event. For each test sentence, even when several words were ambiguous, the model performed correctly.

Concept instantiation works similarly. Though the word cues a number of more specific concepts, only one fits the context. Again, the constraints from the word and from the context combine to produce a unique, specific interpretation of the term. Depending upon the sentence, however, the context may only partially constrain the interpretation. Such is the case in "The teacher kissed someone." "Someone" could refer to any of the four people found in the corpus. Since, in the network's experience, females only kiss males, the context constrains the interpretation of "someone" to be either the busdriver or the pitcher, but no further. Consequently, the model activates the male and person features of the patient while leaving the units representing busdriver and pitcher partially and equally active. In general, the model is capable of inferring as much information as the evidence permits: the more evidence, the more specific the inference.

Finally, sentences in the role elaboration category test the model's ability to infer thematic roles not mentioned in the input sentence. For example, in "The teacher ate the soup," no instrument is mentioned, yet a spoon can be inferred. Here, the context alone provides the constraints for making the inference. Extra roles that are very likely are inferred strongly. When the roles are less likely, or when more than one concept can fill a role, the concepts are only weak inferred. Again, the model processed each test sentence correctly.

Conclusions

Parallel Distributed Processing models have a number of qualities that make them useful as models of language comprehension. First, they allow the simultaneous processing of many constraints. Each bit of relevant information can be applied to a computation. This feature allows far greater interaction between constraints in computing an interpretation of a sentence. As Marcus (1980) points out, the competition and cooperation among constraints of varying strength is an essential aspect of comprehension. Second, PDP models allow the strength of constraints to vary on a continuum rather than discretely. Information may be more or less reliable, and correlations may be more or less strong. The activation level of a unit allows the quality of the information to be represented explicitly and combined effectively. This feature allows *quantitative* interaction among constraints. Third, PDP models naturally perform pattern completion. Given some input, the models add default information to produce an elaborated representation. Importantly, this default information is tailored to the specific input presented. This feature allows context specific inferences about the input to be computed.

Because PDP models learn, useful and complex constraints develop of their own accord. This learning process is slow, requiring a great deal of practice. While comprehension involving strong and regular constraints are learned relatively rapidly, irregular and complex constraints are only learned very slowly. Early on, therefore, the network begins to perform passably, and it slowly improves to correctly process more

sentences.

The model handles ambiguity robustly by processing the input in terms of constraints on a conceptual representation of an event. Within this framework, word disambiguation and concept instantiation are similar processes. Both ambiguous words and vague terms provide conflicting constraints on their interpretation. The constraints from the word itself and additional constraints from its context provide evidence, respectively, for the general and specific features of the concept referred to. Role elaboration is similar, but at the level of interpreting the sentence as a whole. The features of the sentence provide constraints on the unspecified features of the event, such as implicit thematic roles like location, instrument, and manner. When more than one role/filler is likely, each is represented according to its conditional probability.

The interpretation of a sentence evolves as each word is presented to the network. The model sequentially processes each word and immediately adjusts its interpretation of old information and creates expectations of additional information as it attempts to represent the entire conceptual event.

The model cannot, however, represent sentences with embedded clauses. Extending the model to represent more complex sentences is an important goal. Meanwhile, we have learned a great deal about viewing sentence comprehension as a process of weak constraint satisfaction.

References

Carpenter, P. A. & Just, M. A. (1977). Reading comprehension as the eyes see it. In M. A. Just & P. A. Carpenter (Eds.), *Cognitive processes in comprehension*. Hillsdale, NJ: Erlbaum.

Fillmore, C. J. (1968). The case for case. In E. Bach & R. T. Harms (Eds.), *Universals in linguistic theory*. New York: Holt, Rinehart, & Winston.

Hinton, G. E. (1987). Connectionist learning procedures. Tech report #CMU-CS-87-115.

Jordan, M. I. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. Paper presented to the 8th Annual Conference of the Cognitive Science Society. Amherst, MA.

MacWhinney, B. (1987). Competition. In B. MacWhinney (Ed.), *Mechanisms of language acquisition: The 20th annual Carnegie symposium on cognition*. Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Marcus, M. P. (1980). *A theory of syntactic recognition for natural language*. Cambridge, MA: MIT Press.

Marslen-Wilson, W. & Tyler, L. K. (1980). The temporal structure of spoken language understanding. *Cognition, 8*, 1-71.

McClelland, J. L. & Kawamoto, A. H. (1986). Mechanisms of sentence processing: Assigning roles to constituents. In J. L. McClelland, D. E. Rumelhart, and the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 2*. Cambridge, MA.: MIT Press.

Rumelhart, D. E., Hinton, G. E., Williams, R. J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, and the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1*. Cambridge, MA.: MIT Press.

St. John, M. F. & McClelland, J. L. (1987). Reconstructive memory for sentences: A PDP approach. Paper presented to the Ohio University Inference Conference, *Proceedings Inference: OUIC 86*. University of Ohio, Athens, OH.

Recursive Auto-Associative Memory: Devising Compositional Distributed Representations

Jordan Pollack

Computing Research Laboratory
New Mexico State University

INTRODUCTION

A major outstanding problem for connectionist models is the representation of variable-sized recursive and sequential data structures, such as trees and stacks, in fixed-resource systems. Such representational schemes are crucial to efforts in modeling high-level cognitive faculties, such as Natural Language processing. Pure connectionism has thus far generated somewhat unsatisfying systems in this domain, for example, which parse fixed length sentences (Cottrell, 1985; Fanty 1985; Selman, 1985; Hanson & Kegl, 1987), or flat ones (McClelland & Kawamoto, 1986).[1]

Thus, one of the main attacks on connectionism has been on the inadequacy of its representations, especially on their lack of compositionality (Fodor & Pylyshyn, 1988).

However, some design work has been done on general-purpose distributed representations with limited capacity for sequential or recursive structures. For example, Touretzky has developed a coarse-coded memory system and used it both in a production system (Touretzky & Hinton, 1985) and in two other symbolic processes (Touretzky, 1986ab). In the past-tense model, Rumelhart and McClelland (1986) developed an implicitly sequential representation, where a pattern of well-formed overlapping triples could be interpreted as a sequence.

Although both representations were successful for their prescribed tasks, there remain some problems.

- First, a large amount of human effort was involved in the design, compression and tuning of these representations.
- Second, both require expensive and complex access mechanisms, such as pullout networks (Mozer, 1984) or clause-spaces (Touretzky & Hinton, 1985).
- Third, they can only encode structures composed of a fixed tiny set of representational elements, (i.e. like triples of 25 tokens), and can only represent a small number of these element-structures before spurious elements are introduced[2]. These representational spaces are, figuratively speaking, like a "prairie" covered in short bushes of only a few species.
- Finally, they utilize only binary codes over a large set of units.

The compositional distributed representations devised by the technique to be described below demonstrate somewhat opposing, and, I believe, better properties:

- Less human work in design by letting a machine do the work,
- Simple and deterministic access mechanisms,
- A more flexible notion of capacity in a "tropical" representational space: a potentially very large number of primitive species, which combine into tall, but sparse structures.
- Finally, the utilization of analog encodings.

The rest of this paper is organized as follows. First, I describe the strategy for learning to represent stacks and trees, which involves the co-evolution of the training environment along with the access mechanisms and distributed representations. Second, I allude to several experiments using this strategy, and provide the details of an experiment in developing representations for binary syntactic trees. And, finally, some crucial issues are discussed.

RECURSIVE AUTO-ASSOCIATIVE MEMORY

Learning To Be A Stack

Consider a variable-depth stack of L-bit items. For a particular application, both the set of items (i.e. a subset of the 2^L patterns) and the order in which they are pushed and popped are much more constrained than, say, all possible sequences of N such patterns, of which there are 2^{LN}. Given this fact, it should be possible to build a stack with less than LN units (as in a shift-register approach) but more than L units with less than N bits of analog resolution, as in an approach using fractional encodings such as the one I used in the construction of a "neuring machine" (Pollack, 1987a).

[1] It is possible to get around some of these problems by resorting to hybrid modeling, e.g. (Waltz & Pollack, 1985).

[2] Rosenfeld and Touretzky (1987) provide a nice analysis of coarse-coded symbol memories.

The problem is finding such a stack for a particular application.

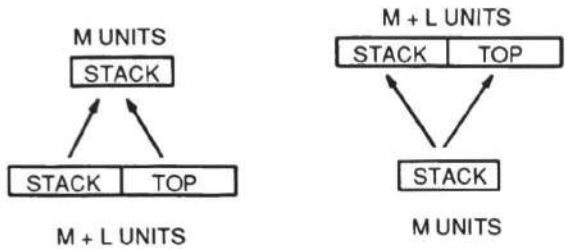


Figure 1.

Proposed inverse stack mechanisms in single-layered feedforward networks.

Consider representing a stack in a activity vector of M bounded analog values, where $M>L$. Pushing a L-bit vector onto the stack is essentially a function that would have $L+M$ inputs, for the new item to push plus the current value of the stack, and M outputs, for the new value of the stack. Popping the stack is a function that would have M input units, for the current value of the stack, and $L+M$ output units, for the top item plus the representation for the remaining stack. Potential mechanisms in the form of single-layered networks are shown in figure 1. The operation performed by a single layer is a vector-by-matrix multiplication and then a non-linear scaling of the output vector to between 0 and 1 by a logistic function.

All we need for a stack mechanism then, are these two functions plus a distinguished M-vector of numbers, ε, the *empty* vector. To push elements onto the stack, simply encode the element plus the current stack; to pop the stack, decode the stack into the top element and the former stack. Note that this is a recursive definition, where previously encoded stacks are used in further encodings. The problem is that it is not at all clear how to design these functions, which involve some magical way to *recursively* encode $L+M$ numbers into M numbers while preserving enough information to consistently decode the $L+M$ numbers back.

One clue for how to do this comes from the Encoder Problem (Ackley, Hinton, & Sejnowski, 1985), where a sparse set of fixed-width patterns are encoded into a set of patterns of smaller width. Back-propagation has been quite successful at this problem, when used in an unsupervised autoassociative mode on a three layer network. Rumelhart, Hinton, & Williams (1986) only demonstrated an 8-3-8 encoder,[3] but Cottrell, Munro, & Zipser (1987) demonstrated a 64-16-64 encoder and Hanson & Kegl (1987) used a 270-45-270 network.

These encoder networks are not directly applicable to the stack mechanism, because the compressed representations are never further processed. But they can be.

Consider a set of training examples for a stack as snapshots of the deepest states some procedure using that stack creates. For example, if the procedure performed the following stack operations generating the corresponding stack states:

operation	stack state
PUSH A	(A)
PUSH B	(B A)
PUSH C	(C B A)
POP	(B A)
PUSH D	(D B A)
PUSH C	(C D B A)
POP	(D B A)
POP	(B A)
POP	(A)
PUSH D	(D A)
POP	(A)
POP	()

Then the deepest states created are (C B A), (C D B A), and (D A), since all the other stack states are "substacks" of these three.

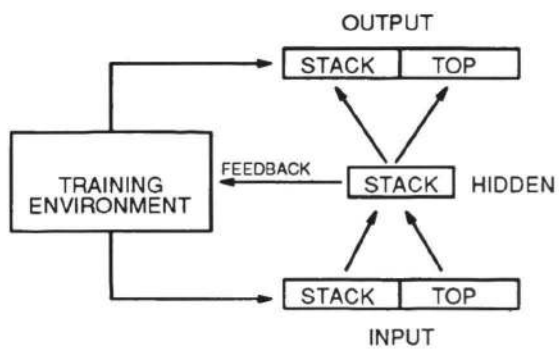


Figure 2.

Recursive Autoassociative Memory. The memory develops compositional distributed representations as the outputs of the hidden layer. The developing representations are fed back into the training environment, which therefore evolves with the weights in the network.

Consider simultaneously training the push and pop mechanisms from figure 1. Taken together, they form an encoder network as shown in Figure 2, with $L+M$ input units, M hidden units, and $L+M$ output units. If the symbols

[3] The three numbers correspond to the number of units in each layer of a 3-layer feed-forward network. I will not describe back-propagation here, assuming that the reader is familiar with the technique. Any programmed implementation of it can be simply modified to do recursive auto-association. These modifications, which involve using two error tolerances, and a mechanism to save, restore, and copy the output values of the hidden layer into the input layer, will be made obvious through the discussion.

above, A through D, represent L-bit vectors, and ε is a distinguished vector of size M, then a successful application (to be defined below) of back-propagation over this network with the following set of training examples can develop the mechanisms for this stack. The roman letters indicate particular L-bit patterns, $R_i(t)$ represents the M values of the hidden units for a particular example during a particular training epoch, and + **is used to indicate concatenation rather than addition**.

input		hidden		output
$A+\varepsilon$	$\rightarrow$	$R_A(t)$	$\rightarrow$	$A'+\varepsilon'$
$B+R_A(t)$	$\rightarrow$	$R_{BA}(t)$	$\rightarrow$	$B'+R_A(t)'$
$C+R_{BA}(t)$	$\rightarrow$	$R_{CBA}(t)$	$\rightarrow$	$C'+R_{BA}(t)'$
$A+\varepsilon$	$\rightarrow$	$R_A(t)$	$\rightarrow$	$A'+\varepsilon'$
$B+R_A(t)$	$\rightarrow$	$R_{BA}(t)$	$\rightarrow$	$B'+R_A(t)'$
$D+R_{BA}(t)$	$\rightarrow$	$R_{DBA}(t)$	$\rightarrow$	$D'+R_{BA}(t)'$
$C+R_{DBA}(t)$	$\rightarrow$	$R_{CDBA}(t)$	$\rightarrow$	$C'+R_{DBA}(t)'$
$A+\varepsilon$	$\rightarrow$	$R_A(t)$	$\rightarrow$	$A'+\varepsilon'$
$D+R_A(t)$	$\rightarrow$	$R_{DA}(t)$	$\rightarrow$	$D'+R_A(t)'$

There are several nonobvious things going on here.

First of all, the (initially random) values of the hidden units, $R_i(t)$, are part of the training environment. As the weights in the system evolve, so does the training environment. The stability and convergence of the network is thus sensitive to the learning rate. It must be set low enough that the change in the hidden representations does not invalidate the decreasing error granted by the change in weights.

In lieu of a formal proof of convergence, I simply argue inductively that the terminal training patterns are constant, so that the patterns of depth 1 become constant as learning proceeds; as they become stable, the patterns of depth 2 start to stabilize, and so on. This paper focuses on the strategy for finding these representations, so while stability and convergence in learning is an important problem, here it is only a secondary issue. As will be discussed later, it may be possible to solve auto-association problems directly through algebraic means.

The second nonintuitive point is that the network is not really being trained as a stack! Considering that the input-to-hidden function performs a PUSH, and the hidden-to-output function function a POP, the training regime involves multiple PUSHes, but only single POPs.

Enter the notion of *successful training*, by which I mean the usual termination condition for back-propagation: For each example, and for each unit, the absolute value of the difference between desired and actual output is less than some tolerance.

For the encoder and decoder to really work, however, the tolerance on the developed representations has to be quite sharp. After pushing A, the stack is represented by R_A. After pushing B, the stack is represented by R_{BA}. Popping R_{BA} returns B′ and $R_A{}'$. In order to successfully pop $R_A{}'$ to get A′ and ε', $R_A{}'$ must be very similar to R_A.

In running the simulations, then, I use two different tolerances, a terminal tolerance, τ, for the input/output bit-vectors, and another non-terminal tolerance, ν for the developing hidden representations. In the experiments described below, I use τ=0.2 and ν=0.05.

The third conceptual problem is that the two parts of the RAAM really define separate mechanisms, a PUSH encoder and a POP decoder, which will be embedded in different parts of an application network with some simpler sort of memory and control logic. That these mechanisms are tightly coupled in training does not mean that they always form a single network.

I have run several experiments in learning distributed representations for sequences whose details are omitted for lack of space. One 21-12-21 RAAM learned to be a stack for a network using a recursive subgoal strategy to solve the Towers-of-Hanoi Puzzle; Another 15-10-15 RAAM learned to represent sequences of letters (encoded as 5 bit numbers) in common words; and a third experiment on a 4-3-4 RAAM trained on all eight depth-3 patterns of a single bit stack developed into a shift register.

Learning to Represent Trees

Considering that a stack is really a right-branching binary tree with a distinguished empty symbol, it should be obvious that this mechanism can also be adapted for dealing with other *fixed valence* trees. For a training set consisting of unlabeled binary trees where the terminals are K-bit binary vectors, a three-layer network with $2K$ input units, K hidden units, and $2K$ output units can be used to develop representations for a set of such trees. The input-to-hidden function encodes two trees into a new, higher-level, tree, while the hidden to output function decodes a tree into two subtrees.

Consider the tree, ((D (A N))(V (P (D N))), as one member of a training set of such trees. If the aforementioned network is successfully trained with the following patterns (among other such patterns in the training environment), the resultant encoder and decoder networks can reliably form representations for these binary trees.

input	hidden	output
A+N	$\rightarrow R_{AN}(t)$	$\rightarrow$ A′+N′
D+$R_{AN}(t)$	$\rightarrow R_{DAN}(t)$	$\rightarrow$ D′+$R_{AN}(t)'$
D+N	$\rightarrow R_{DN}(t)$	$\rightarrow$ D′+N′
P+$R_{DN}(t)$	$\rightarrow R_{PDN}(t)$	$\rightarrow$ P′+$R_{DN}(t)'$
V+$R_{PDN}(t)$	$\rightarrow R_{VPDN}(t)$	$\rightarrow$ V′+$R_{PDN}(t)'$
$R_{DAN}(t)$+$R_{VPDN}(t)$	$\rightarrow R_{DANVPDN}(t)$	$\rightarrow R_{DAN}(t)'$+$R_{VPDN}(t)'$

DETAILS OF AN EXPERIMENT

In fact, the above example was part of the first experiment I ran on learning to represent trees. Consider a simple grammar, where every rule expansion has exactly two components:

S -> NP VP | NP V
NP -> D AP | D N | NP PP
PP -> P NP
VP -> V NP | V PP
AP -> A AP | A N

Given a set of strings in the language defined by this grammar, any old parser is capable of returning bracketed binary trees which will make up our training set. I made up such a set of strings, and used a parser to get the following set of trees:

(D (A (A (A N))))
((D N)(P (D N)))
(V (D N))
(P (D (A N)))
((D N) V)
((D N) (V (D (A N)))))
((D (A N)) (V (P (D N))))

Each terminal (D A N V & P) was represented as a 1-bit-in-5 code padded with 5 zeros. A 20-10-20 RAAM devised the representations shown in figure 3.

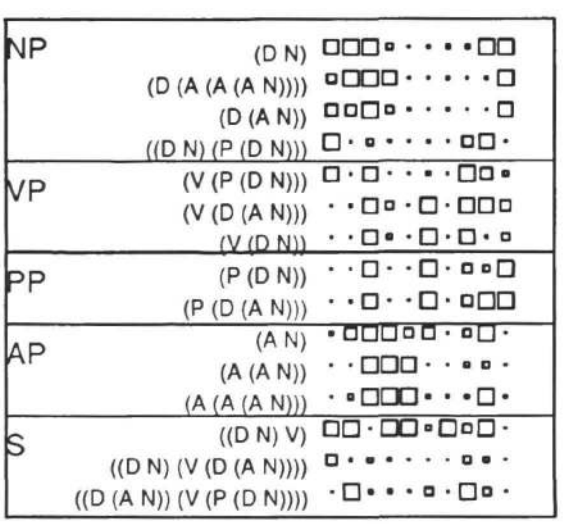


Figure 3.

Representations of all the binary trees in the training set, devised by a 20-10-20 RAAM, manually clustered by phrase-type. The squares represent values between 0 and 1 by area.

I labeled each tree and its representation by the phrase type in the grammar, and sorted them by type. The RAAM has clearly developed a representation with similarity between members of the same type. For example, the third feature seems to be clearly distinguishing sentences from non-sentences, the fifth feature seems to be involved in separating adjective phrases from others, while the tenth feature appears to distinguish prepositional and noun phrases from others.[4]

At the same time, the representation must be keeping enough information about the subtrees in order to accurately reconstruct them.

The encoder and decoder networks thus form a recursive well-formedness test as follows: Take two trees, encode them into a new, higher-level, tree, and decode that back into two subtrees. If the subtrees are reconstructed within tolerance (i.e. τ for terminal binary vectors, and ν for non-terminal analog vectors) than that tree can be said to be well-formed.

The implications for conventional parsing is that instead of searching a grammar for a rule whose right-hand sides matches the two phrase-markers, we can accomplish this search with a simple feed-forward computation and parallel element-by- element comparison. This could lead to a somewhat nifty speed improvement.

The implications for connectionist parsing is that the output of an unsupervised RAAM could be the teacher for a supervised sequential learning technique such as sequential cascaded networks (Pollack, 1987b).

Furthermore, the well-formedness test can be used in a generator as follows. Start with a pool of trees composed of the terminals. Take every pair of trees from the pool, apply the well-formedness test, and if it passes, add the new higher-level tree to the pool.

Running this generator over the network formed from the above experiment yielded the following two interesting parse trees, among many silly ones:

(((D N)(P (D N)))(P (D N)))
((D (A N))(V (D N))

The first seems to be a recursive application of the NP -> NP PP rule, while the second is a single instance of a known NP newly appearing in a VP.

[4] In fact, by these metrics, the test case ((D N)(P (D N))) should really be classified as a sentence; since it was not used in any other construction, there was no reason for the RAAM to believe otherwise.

In my experiments thus far, RAAMS have only shown evidence of very limited generativity in this sense. The issue will be discussed further below.

DISCUSSION

The Tricks That Make RAAMS Work

Extensional Programming (Cottrell, et. al., 1987). Given that, with suitable control logic, the encoder produces a fixed-length analog representation for a sequence or tree which the decoder can decompose and reconstruct into a good facsimile of the original sequence or tree, then the fixed-length vector **must** be representing that sequence or tree.

Embedding the developing representations into the training environment. This is really new, though McClelland & St. John (1986), apparently tried something similar.

Recursively Reduced Descriptions. Recursive Autoassociative Memory appears to implement one of Hinton's (unpublished) idea's that has been floating around for a couple of years. With implementation, however, comes both validation and better understanding.

Constrained Co-evolution of Multiple Mechanisms. The encoder and decoder of a RAAM are really separate mechanisms which are evolved simultaneously and cooperate to develop a shared representation. There may be a useful foundational principle at work

Performance Induced from Tolerance Chaining. Neither stack nor tree mechanisms are really trained to be what they become. I am making an induction, backed up by lots of experimentation, that if an original encoded representation is compressed even further by the encoder, but can be reconstructed by the decoder to within a close tolerance of the original, then this reconstructed representation can be decoded almost as well as the original. It stands to reason that the error tolerance must get smaller as the desired representations get deeper, but this has not yet been quantified.

Compositionality is Settled

Fodor & Pylyshyn (1988) attacked the non-compositional nature of connectionist representations, while McDermott (1986) challenged the community to be able to represent and reason about the meaning of a complex sentence like: "She is more at home with her fellow students than with me, her advisor."

Compositionality and complexity are no longer crippling issues for connectionism. The representations that RAAM's develop may not be obvious, except in the simplest and most highly constrained cases, but they are compositional in the strictest sense.

Systematicity is not Settled

Unfortunately, Fodor & Pylyshyn's generative capacity issue, which they called "systematicity", still stands for the time being. There are essentially three possibilities, which will be explored in future work. Given that I have shown some limited generative capacity in the form of a small number of new useful representations, I believe that either the second, third, or both will be the case.

1. RAAM will not be able to develop systematicity;
2. Better training environments and better activation functions are needed; or
3. Critical mass, in terms of network size or training environment size, is needed.

Default Topology and Training is not Essential

There is no reason why only single level networks for the encoder and decoder should be used. Hidden units in the decoder as well as real thresholds will probably be necessary for a RAAM to develop a real analog stack representation, which may affect generative capacity[5].

On the other hand, this simple form of autoassociation may be directly solvable by linear methods: Cottrell, et. al. (1987) reported that their autoassociator essentially performs decomposition into principal components, while Bourlard & Kamp (1988) reported that autoassociative networks have direct and optimal algebraic solutions. Whether such methods can be implemented under the constraints of extreme locality, and whether they might extend to recursive autoassociation are still, however, open questions.

CONCLUSIONS

Implications: The Ultimate Capacity Of RAAMS

I do not have any closed analytical forms for the capacity of RAAMs. Somebody else will have to do that work. Given that is is not really an file-cabinet or content-addressable memory, but a memory for a gestalt of rules for recursive pattern compression and reconstruction, results such as Willshaw's (1981) and Hopfield's (1982) do not directly apply. Binary patterns are not being stored, so one cannot simply count how many.

I have considered the capacity of such a memory in the limit, however, where the actual functions and analog representations are not bounded by single linear transformations and sigmoids or by 32-bit floating point resolution.

[5] The ultimate irony here would be if the competence/performance distinction reduced to a problem in roundoff error.

Saund (1987) has investigated autoassociation as a method of *dimensionality reduction*, and asserted that, in order to work, the data must be constrained to form a small dimensional parametric surface in a larger dimensional space.

Consider just a 2-1-2 autoassociator. It is really a reconstructible mapping from data points on the unit square to unique points on the unit line. In order to work, the environment should define a parametric 1-dimensional curve in 2-space, perhaps a set of connected splines. As more and more data points need to be encoded, this parametric curve must get "curvier" to cover them. In the limit, it is no longer a 1-dimensional curve, but a space-filling curve with a fractional dimension. (At this point, however, the similarity between nearby points is lost.)

It is not yet the time to discuss the implications of *fractal representations* for AI: Just consider that the decomposition of meaning cannot really stop at a semantic primitive such as "MTRANS".

Applications: Inference By Association

If the reader almost believes that (1) complex AI-style representations can now be encoded into fixed-length analog vectors, which are compositional, similarity-based, and recursive in nature, and (2) that functions can be devised, say, using back-propagation, which perform arbitrary associations (that can generalize) between such fixed-width vectors, than it is not too hard a leap to believe that it may be possible, in the near future, to construct simple, constant-time, deterministic (but fallible) mechanisms for apparently rule-based processes such as parsing, logical deduction, plausible inference, and, perhaps, even, syntactic transformations.

I will be exploring several of these applications, with the slogan of "Associative Inference" over the next year.

ACKNOWLEDGEMENTS & REFERENCES

Tony Plate implemented back-propagation and the necessary modifications in C in exchange for a nice lunch; another is due for solving the traveling scientist problem (filling in the blanks of many references).

Ackley, D. H., Hinton, G. E. & Sejnowski, T. J. (1985). A learning algorithm for Boltzmann Machines. *Cognitive Science*, 9, 147-169.

Bourlard, H. & Kamp, Y. (1988) Auto-Association by Multilayer Perceptions and Singular Value Decomposition Technical Report M217. Brussels: Phillips Research Laboratory.

Cottrell, G. W. (1985). Connectionist Parsing. In *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*. Irvine, CA.

Cottrell, G., Munro, P., and Zipser, D. (1987). Learning Internal Representations from Gray-Scales Images: An Example of Extensional Programming. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*. Seattle, WA. 461-473.

Fanty, M. (1985). Context-free parsing in Connectionist Networks. TR174, Rochester, N.Y.: University of Rochester, Computer Science Department.

Fodor J. & Pylyshyn, Z. (1988) Connectionism and Cognitive Architecture: A critical Analysis. *Cognition*, 28, 3-71.

Hanson, S. and Kegl, J. (1987) PARSNIP: A connectionist network that learns natural language grammar from exposure to natural language sentences. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, Seattle, WA., 106-119.

Hinton G. (unpublished). Representing Part-Whole Hierarchies in Connectionist Networks.

Hopfield, J. J. (1982). Neural Networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79, 2554-2558.

McClelland J. & St. John, M. (1986.) Reconstructive Memory for Sentences: A PDP Approach. *Proceedings of the Ohio University Inference Conference*.

McClelland, J. and Kawamoto A. (1986) Mechanisms of Sentence Processing: Assigning Roles to Constituents. In J. L. McClelland, D. E. Rumelhart & the PDP research group, (Eds.), *Parallel Distributed Processing: Experiments in the Microstructure of Cognition*, Vol. 2. Cambridge: MIT Press

McDermott, D. (1986) What AI needs from connectionism. Appendix to McClelland, J.L., Feldman, J.A., Bower, G. & McDermott, D. Connectionist models and cognitive science: goals, directions and implications. A report on an NSF workshop.

Mozer, M. (1984) Inductive Information Retrieval Using Parallel Distributed Computation. Technical Report. La Jolla: Institute for Cognitive Science, UCSD.

Pollack, J. B. (1987a) *Connectionism and natural language processing*. Cambridge: MIT Press. (In preparation; Available as MCCS-87-100, Computing Research Laboratory, NMSU, Las Cruces)

Pollack, J. B. (1987b) Cascaded back-propagation on dynamic connectionist networks. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, Seattle, WA.

Rosenfeld R. & Touretzky, D. (1987). Four capacity models for coarse-coded symbol memories. CMU-CS-87-182. Pittsburgh: Computer Science Dept., Carnegie-Mellon University.

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.

Rumelhart, D. E. & McClelland, J. L. (1986). On Learning the Past Tenses of English Verbs. In J. L. McClelland, D. E. Rumelhart & the PDP research group, (Eds.), *Parallel Distributed Processing: Experiments in the Microstructure of Cognition*, Vol. 2. Cambridge: MIT Press.

Saund, E. (1987) Dimensionality Reduction and Constraint in Later Vision.*Proceedings of the Ninth Annual Conference of the Cognitive Science Society,*, 908-915, Seattle, WA.

Selman, B. (1985). Rule-Based Processing in a Connectionist System for Natural Language Understanding. CSRI-168, Toronto, Canada: University of Toronto, Computer Systems Research Institute.

Touretzky, D. S. & Hinton, G. E. (1985). Symbols among the neurons: details of a connectionist inference architecture. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence.* Los Angeles, CA.

Touretzky, D. S. (1986a). BoltzCONS: Reconciling connectionism with the recursive nature of stacks and trees. In *Proceedings of the 8th Annual Conference of the Cognitive Science Society.* Amherst, MA, 522-530.

Touretzky, D. S. (1986b). Representing and transforming recursive objects in a neural network, or "trees do grow on Boltzmann machines". In roceedings of the 1986 Institute of Electrical and Electronics Engineers International Conference on Systems, Man, and Cybernetics. Atlanta, GA.

Waltz, D. L. & Pollack, J. B. (1985). Massively Parallel Parsing: A strongly interactive model of Natural Language Interpretation. *Cognitive Science*, 9, 51-74.

Willshaw, D. (1981). Holography, Associative Memory, and Inductive Generalization. In Hinton, G. and J. Anderson (Eds.) *Parallel Models of Associative Memory.* Hillsdale: Lawrence Erlbaum Associates.

Experiments With Sequential Associative Memories

Stephen I. Gallant* and Donna J. King

College of Computer Science
Northeastern University
Boston, Ma. 02115 USA

Abstract

Humans are very good at manipulating sequential information, but sequences present special problems for connectionist models.

As an approach to sequential problems we have examined totally connected subnetworks of cells called *sequential associative memories (SAM's)*. The coefficients for SAM cells are unmodifiable and are generated at random.

A subnetwork of SAM cells performs two tasks:

1. Their activations determine a state for the network that permits previous inputs and outputs to be recalled, and
2. They increase the dimensionality of input and output representations to make it possible for other (modifiable) cells in the network to learn difficult tasks.

The second function is similar to the *distributed method*, a way of generating intermediate cells for non-sequential problems.

Results from several experiments are presented. The first is a robotic control task that required a network to produce one of several sequences of outputs when input cells were set to a corresponding 'plan number'.

The second experiment was to learn a sequential version of the parity function that would generalize to arbitrarily long input strings.

Finally we attempted to teach a network how to add arbitrarily long pairs of binary numbers. Here we were successful if the network contained a cell dedicated to the notion of 'carry'; otherwise the network performed at less than 100% for unseen sequences longer than those used during training.

Each of these tasks required a representation of state, and hence a network with feedback. All were learned using subnetworks of SAM cells.

Keywords: Connectionist Models, Learning, Sequences, Distributed Representation, Robotic Control

1 Introduction

A supervised, discrete SEQUENCE LEARNING PROBLEM involves learning to produce a sequence of correct output vectors

$$C^1, C^2, \cdots, C^k \quad (C^i \in \{+1, -1\}^g)$$

*Partially supported by National Science Foundation grant IRI-8611596. Thanks to Emmanouil Kalfaoglu for help with experiments. The first portion of this paper is revised from [Gallant 1987].

in response to a corresponding sequence of example input vectors

$$E^1, E^2, \cdots, E^k \ (E^i \in \{+1, -1\}^p)$$

where *order counts.*

For example if E^i and C^i are one dimensional vectors so that $E^i, C^i = \pm 1$ then we might desire

$$C^i = \begin{cases} +1 & \text{if } \sum_{j=1}^{i} \{E^j | E^j = +1\} \text{ is odd} \\ -1 & \text{otherwise.} \end{cases}$$

This is clearly a sequential version of the parity problem for a network with one input cell and one output cell. Note that the previous context ($\{E^j | j < i\}$) is important; knowing the value of E^i alone is not enough information to determine C^i.

There are many interesting sequential learning problems from a variety of disciplines: speech understanding (even for a single word), language understanding, robotic control (i.e. coordinated muscular activities that take advantage of sensory input), sequential fault detection problems, dynamic vision tasks, end even static vision problems if a single scene is analyzed by a sequence of visual fixations on parts of that scene.

Sequences can be difficult for connectionist models, particularly if they are of indefinite length. For some tasks (e.g. pronoun disambiguation) we cannot specify an n where the n previous inputs are always sufficient to allow correct outputs. Moreover if we remember the "window" of n previous inputs and apply brute force methods to every possible n-input configuration then the combinatorics becomes devastating. Therefore some sort of processing is required to extract and remember relevant information from prior context.

Recently there has been increased attention to sequence processing by connectionist models. Jordan [Jordan 1986a,Jordan 1986b] has employed a simple subnetwork of cells to represent the state of a network by encoding previous activations from each output cell in the activation value of a corresponding state cell using low order binary decimal bits. Thus if $u_i(T)$ is the activation of output cell u_i at time T then a corresponding state cell, ${u_i}^S(T)$ would have activation

$${u_i}^S(T) = \sum_{t=0}^{T} (0.5)^{T-t} u_i(t).$$

This method preserves information on *all* previous outputs but at the (inevitable) cost of requiring manipulation of high precision activations.

In this paper we explore highly connected subnetworks with feedback called *sequential associative memories* or *SAM*'s. SAM's are composed of linear discriminant cells with fixed, randomly generated weights. These subnetworks serve two important functions:

1. Their activations define a state for the network that allows previous inputs and outputs to affect the current output.

2. They help trainable cells elsewhere in the network to solve difficult (i.e. nonseparable) computational tasks requiring intermediate (or "hidden") cells.

SAM cell activations are discrete, so there is no problem with high precision activation values. However since a finite number of bits can only encode a finite amount of information, the SAM

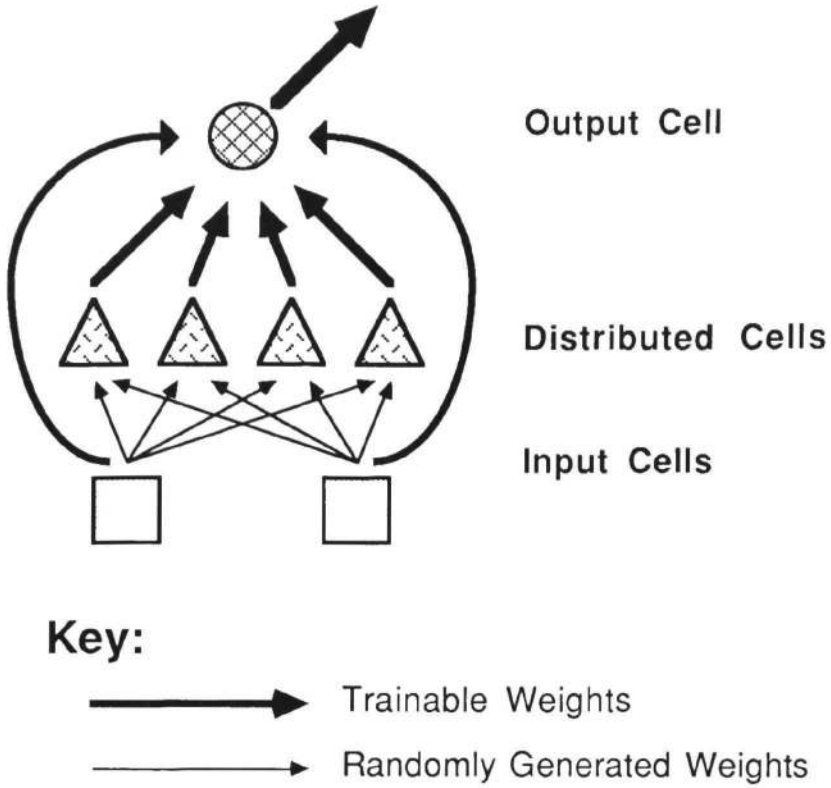


Figure 1: The Distributed Method.

cells cannot have perfect memory; they must 'forget' some information when their capacity is exceeded.

The following sections give a motivation for, and a more precise definition of, sequential associative memories and describe their role in several learning experiments.

2 The Distributed Method

We have previously described a technique called the *distributed method* for creating intermediate cells so that a network could learn nonseparable functions [Gallant 1986a, Gallant & Smith 1987]. See figure 1.

In this method all of the network cells (except input cells) are linear discriminants [Fisher 1936] (also known as threshold logic units [McCulloch & Pitts 1943] or perceptrons [Rosenblatt 1961]). All cells compute discrete activation values of $+1$, -1, or 0 according to whether the weighted sum of their inputs (plus a constant bias term) is > 0, < 0, or $= 0$ respectively. The weights for distributed cells are chosen at random[1] and remain fixed. Output cell weights are generated using the pocket algorithm [Gallant 1986b], a modification of perceptron learning.

Assume there are more distributed cells than input cells. For every setting of the input cells the activations of the distributed cells will form a distributed representation [Rumelhart & McClelland 1986, chap. 3] of the input that lies in a higher dimensional space than the original input. This higher dimensional representation eases the learning task for the output cells and, in

[1] We use integers between -5 and $+5$, but random real values between -1 and $+1$ are arguably better.

particular, makes it more likely that the transformed problem will be separable.

For n training examples our experiments have shown that adding between $\frac{1}{4}n$ and n distributed cells is usually sufficient to make (non-contradictory) data separable for the most difficult classical problems such as parity. Fewer suffice for more 'natural' problems. This bound is roughly consistent with a theoretical result by Cover [Cover 1965] for random functions. By restricting distributed cells to linear discriminants, however, we are also able to preserve robustness because a new input that is close to an example used in training the network is likely to produce the same output as that example. Had we used arbitrary random functions for distributed cells rather than restricting ourselves to the subclass of random *linear discriminants*, the increased probability of separability would have remained but the robustness would have been lost. For a fuller description of the distributed method see [Gallant & Smith 1987].

3 Sequential Associative Memories

The distributed method increases learning ability while preserving robustness (in the sense described), but this technique gives no help with sequences since the network is unaffected by previous inputs and outputs. What is missing is feedback and the notion of state to capture previous behavior.

In figure 2 we see a generalization of the distributed method where the added cells also receive inputs from previous network outputs and from each other. We call such a subnetwork a *sequential associative memory (SAM)*. The state of the network is the pattern of activations in the SAM cells, and it is influenced by previous inputs, outputs, and states. Moreover the SAM cells aid learning by the trainable output cells while still preserving robustness, just as with the distributed method.

3.1 Network Dynamics

The dynamic operation of the network for sequences is as follows:

1. Initialize SAM cell activations and output cell activations to 0.
2. Set the activations of input cells to the {first / next} set of inputs in a sequence.
3. For each SAM cell, compute its new activation value but do not change that activation until all other SAM cells have computed their new values. After computing the new activation values for all SAM cells, modify all the activations accordingly.
4. For each output cell, compute its new activation value and immediately change its activation to that value. These activations are the {first / next} network outputs for the sequence.
5. Go to 2.

Note that all SAM cells can recompute and update their activations in parallel, making this architecture attractive for parallel hardware implementations.

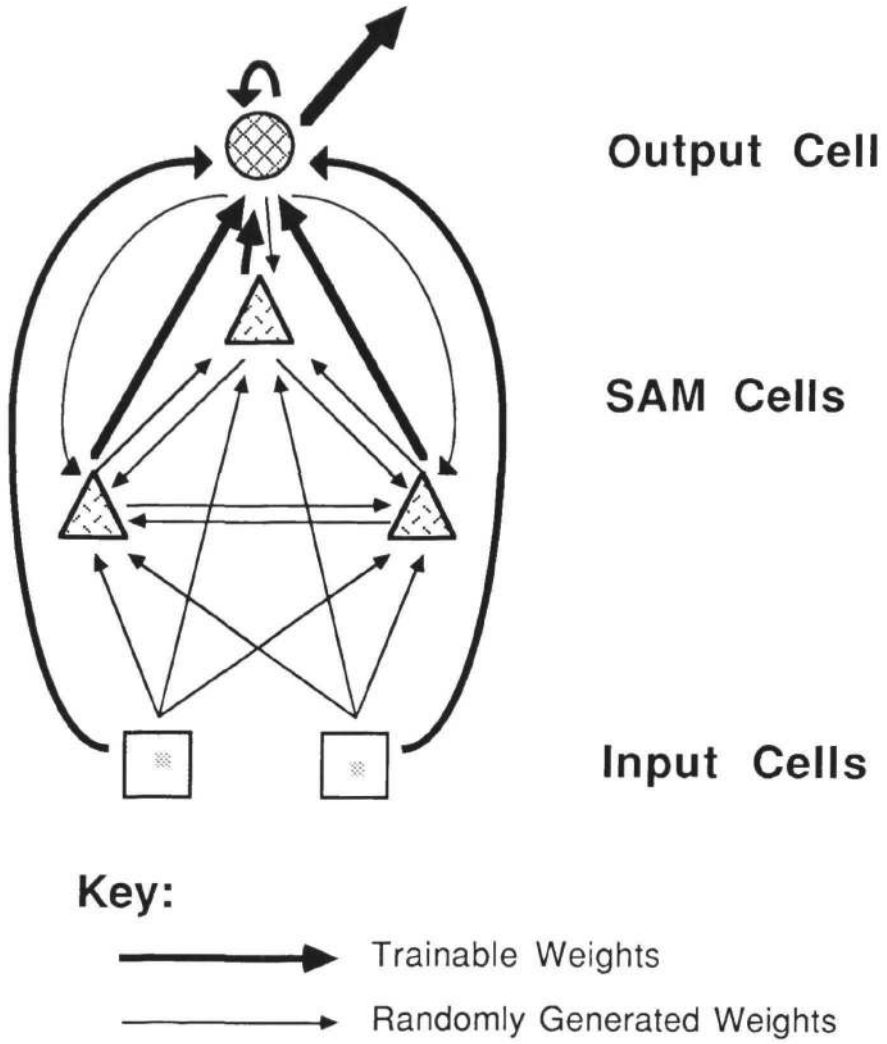


Figure 2: Network with 3 SAM Cells.

3.2 Input Gain

For some tasks it is necessary to multiply activations for input and output cells by a *gain factor* to increase their effect on the SAM cells. If this is not done the SAM cells are not sensitive enough to input and output values and performance can be poor. The higher the gain the more the emphasis is placed upon *recent* inputs and outputs; thus gain might be viewed as a parameter that determines an inductive bias for the learning algorithm. A good setting for the gain factor varies with the number of SAM cells and the problem type.

The easiest time to implement gain is at the time when the (random) SAM connection weights are generated; we simply multiply the weights from input or output cells to SAM cells by the gain factor.

4 Experiments

4.1 Robotic Control

We tried SAM networks with a robotic control problem similar to the one studied by Jordan [Jordan 1986b]. The network had 4 input cells that were set to one of 16 possible 'plan numbers'. Every plan number corresponded to a sequence of outputs from a single output cell over 4 iterations; for example plan $<+1,-1,-1,+1>$ might correspond to the sequence $\{-1,-1,+1,-1\}$. We set the input cells to the same plan number at each time step, and the correspondence between plan numbers and correct output sequences was chosen at random.

Notice that the network cannot determine its correct output based solely upon its (current) input (even if its last output were available). Therefore a 'state' must be maintained by the system and a feedback network is required.

For a network with 4 input cells and 1 output cell we found that 16 sequences of length 4 could be learned reliably by adding 40 SAM cells (using gain 1) in about 5000 iterations (consuming several minutes of CPU time).

We then tried a harder problem involving 5 input cells and 32 sequences of length 5. Here about 150 SAM cells with gain 1 sufficed for reliable 100% performance after about 8000 iterations.

Finally we tried problems with 6 input cells and 64 sequences of length 6. Here 350 SAM cells with gain 1 were required to reliably give 100% performance after about 4000 iterations.

Our conclusion was that SAM cells allow networks to learn simple, sequential robotic tasks in reasonable time.

It is interesting to analyze this task from an information theory viewpoint. If we consider the random coefficients of SAM cells as carrying no information (since they are not specific to the particular set of plans and output sequences being represented) then all information is carried in the trainable weights leading to output cells. The number of such weights equals the number of cells in the network. In our experiments the number of weights was approximately the same as the number of output bits that were to be learned, i.e. the number of plans multiplied by the length of each sequence. For example with 5 input cell sequences and 150 SAM cells there are $5 + 150 + 1$ trainable weights and these can correctly produce 32×5 total output bits. The magnitudes of the weight values did not appear to grow with the difficulty of the problem.

The total number of output bits gives a lower bound for the encoding size of this task; therefore the output cell weights appear to give reasonably efficient encodings because they seem to be growing linearly with the lower bound.

4.2 Sequential Parity

For our second experiment we set out to teach a network the concept of parity as a sequential task. Our goal was for the network to compute parity *for arbitrarily long input sequences*, an impossible task for a feed-forward network.

We used a network with one input cell, one output cell and a collection of SAM cells. The problem was made somewhat harder by allowing the output to see only the current input and the SAM activations but not the previous output. We found that only 10 SAM cells (with gain of 10) were sufficient to quickly and reliably learn parity for arbitrary length sequences.[2] Five SAM

[2]We tested the resulting networks on 50 unseen sequences of length 100 to verify that they had learned parity with high, but not 100%, certainty.

cells were usually too few for this task. To our knowledge this is the first connectionist network to learn parity for arbitrary length sequences.

4.3 Sequential Addition

We also attempting to teach a network sequential (right-to-left) addition of pairs of arbitrarily long numbers using networks composed of 2 input cells, 1 output cell and a collection of SAM cells as in figure 2. This is an easy task if we dedicate an extra output cell to the computation of 'carry'; 10 SAM cells usually suffice to quickly learn addition of arbitrary length numbers with perfect generalization [Gallant 1987].

Without such a dedicated carry cell the problem is much harder because the notion of carry must be represented in the collective activations of the SAM cells.

For this problem a network with 300 SAM cells (and gain 300) learned to add a training set consisting of 100 pairs of 40 bit numbers using 2500 iterations.[3] Testing on groups of 20 (unseen) pairs of numbers of various lengths the network generalized as follows:

Length of numbers added:	20	30	40	50	75	100
% Sequences correct:	100%	100%	95%	100%	95%	95%

For a sequence to be counted as correct in the testing we required that *every* output bit be correct. Although the network failed to precisely learn the concept of addition, we considered it encouraging that it was able to generalize from 40-bit pairs of numbers to 75-bit and 100-bit pairs with 95 per cent of the sequences totally correct.

The improvement in generalization over those tests reported in [Gallant 1987] we attribute to two factors: higher gain settings and the use of longer sequences for training. Experiments are continuing on this problem.

5 Concluding Remarks

We have described a connectionist model with feedback and state called sequential associative memories and several experiments involving sequences. The most important current research involving SAM networks is to determine how well they train and generalize for difficult sequential pattern recognition problems such as speech understanding, fault detection, natural language, and possibly vision. While we hold little hope that SAM networks *alone* could tackle these difficult problems, we are more optimistic about their *combination* with other techniques (such as "windows" and unsupervised learning of input features during a pre-processing step).

We believe that the problem of manipulating sequences represents an important challenge for connectionist research and for machine learning in general.

References

[Cover 1965]
Cover, T. M. Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition. IEEE Trans. Electronic Computers, Vol. 14, 326-334, 1965.

[3] We count an iteration as one presentation of a single training sequence, as contrasted to a 'sweep' through all 100 sequences.

[Fisher 1936]
Fisher, R. A. The use of multiple measurements in taxonomic problems. Ann. Eugenics, 7, Part II, 179-188. Also in *Contributions to Mathematical Statistics* (1950) John Wiley, New York.

[Gallant 1986a]
Gallant, S. I. Three Constructive Algorithms for Network Learning. Proc. Eighth Annual Conference of the Cognitive Science Society, Amherst, Ma., Aug. 15-17, 1986.

[Gallant 1986b]
Gallant, S. I. Optimal Linear Discriminants. Proc. Eighth International Conference on Pattern Recognition, Paris, France, Oct. 28-31, 1986.

[Gallant & Smith 1987]
Gallant, S. I., and Smith, D. Random Cells: An Idea Whose Time Has Come and Gone... And Come Again? IEEE International Conference on Neural Networks, San Diego, Ca., Vol. II, 671-678, June 1987.

[Gallant 1987]
Gallant, S. I. Sequential Associative Memories. Technical Report NU-CCS-87-20, Northeastern University College of Computer Science.

[Jordan 1986a]
Jordan, M. I. Attractor Dynamics and Parallelism in a Connectionist Sequential Machine. Proceedings of the Eighth Annual Conference of the Cognitive Science Society, Amherst, Ma., 1986

[Jordan 1986b]
Jordan, M. I. Serial Order: A Parallel Distributed Processing Approach. Institute for Cognitive Science Report 8604, University of California, San Diego, May 1986.

[McCulloch & Pitts 1943]
McCulloch, W. S. & Pitts, W. H. A logical calculus of the ideas imminent in nervous activity. Bulletin of Math. Biophysics, 5, 115-133 (1943).

[Rosenblatt 1961]
Rosenblatt, F. *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms.* Spartan Press, Washington, DC.

[Rumelhart & McClelland 1986]
D. E. Rumelhart & J. L. McClelland (Eds.) *Parallel Distributed Processing: Explorations in the Microstructures of Cognition.* MIT Press.

Representing part-whole hierarchies in connectionist networks

Geoffrey E. Hinton
Computer Science Department & Psychology Department, University of Toronto

Introduction

One reason for scepticism about connectionist models that use distributed representations is that there are currently no convincing demonstrations of how these models can represent complex, articulated structures. Drew McDermott (personal communication) has suggested that the approach would be far more convincing if it could come up with a sensible scheme for representing the meaning of a sentence such as: "She seems to be more at ease with her fellow students than with me, her adviser." This meaning is clearly composed of several major constituents with relationships between them, and each major constituent has its own, complex, internal structure. A representational scheme for dealing with meanings of this complexity must, at the very least, specify how the meanings of whole expressions are related to the meanings of their constituents and how it is possible, in some sense, to have the whole meaning in mind at once.

The example given above is typical of examples from many different domains. It appears that whenever people have to deal with complexity they impose part-whole hierarchies in which objects at one level are composed of inter-related objects at the next level down. In representing a visual scene or an everday plan or the structure of a sentence we use hierarchical structures of this kind. The main issue addressed in this paper is how to represent complex part-whole hierarchies in a connectionist network. Three different methods are described.

Symbols and the conventional implementation of hierarchical structures

It will be helpful to begin by reviewing the standard way of implementing hierarchical data-structures in a conventional digital computer. There are obviously many minor variations, but a suitable paradigm example is the kind of record structure that is found in languages like Pascal (but without the type constraints).

Each instance of a record is composed of a predetermined set of fields (sometimes called "slots" or "roles") each of which contains a pointer to the contents of the field which may be either another instance of a record, or a primitive object. Since the pointers can be arbitrary addresses, this is a very flexible way of implementing a hierarchical data-structure, but the flexibility is bought at the price of the von Neumann bottleneck: The addressing mechanism means that only one pointer can be followed at a time. [1]

The essence of a symbol is this: It is a small representation of an object that provides an "remote access" path to a fuller representation of the same object. [2] For example, the address of a record structure is a small representation and the whole record that it points to is a fuller representation. In general, this fuller representation is itself composed of small representations (e. g. addresses) of the structures that fill the fields of the record. Because a symbol is small, many symbols can be put together to create a "fully-articulated" representation of some larger structure and the size of this fully-articulated represention need not be any larger than the fully-articulated representations of its constituents.

When addresses are used as symbols, there is normally a purely arbitrary relationship between the internal structure of a symbol and the fully articulated representation that it provides access to. Looking at the individual bits in the symbol provides no information about what it represents. Occasionally this is not quite true. If, for example, one type of data-structure is kept in the top half of memory and another type in the bottom half, the first bit of a symbol reveals the type of the data-structure that it provides access to. So it is possible to check the type without following the pointer. This trick can obviously be extended so that many of the bits in

[1] Architectures such as the Connection Machine (Hillis, 1985) use routing hardware that allows many pointers to be followed at once, but even with hardware support, the routing is quite slow.

[2] There is, of course, much debate about the meaning of the word "symbol". The informal definition given here emerged from conversations with Allen Newell.

a symbol convey useful information. A symbol can then be viewed as a "reduced description" of the object.

The conclusion of this paper is that patterns of activity in parts of a connectionist network need to exhibit the double life that is characteristic of symbols. The patterns must allow remote access to fuller representations, but so long as the patterns are also reduced descriptions this remote access need only be used very occasionally (e.g. a few times per second). Most of the processing can be done by parallel constraint-satisfaction on the patterns themselves.

Method 1: The fully-parallel implementation

Perhaps the most obvious way to implement part-whole hierarchies in a connectionist network is to use the connections themselves as pointers. Figure 1 shows an example taken from the work of Rumelhart and McClelland (1981). It is a network that recognizes a word when given partial information about the features of the letters in the word. We use this as the standard, concrete example of a part-whole hierarchy because it has clearly defined levels and the parts have convenient names, but this paper is not about word recognition. Because each relationship in the hierarchical tree-structure is implemented by its own dedicated connection, it is possible to do a lot of parallel processing during recognition. Simultaneously, many different letters can check whether the features they require are present, and many different words can check whether the letters they require are present.

One very important aspect of the Rumelhart and McClelland network is that each of the letter units has to be replicated for each of the four possible positions of a letter within the word (they restrict themselves to four-letter words). There is a separate unit for an H as first letter and an H as second letter. All the letter features and all the knowledge about which combinations of letter features make an H must also be replicated for each of the four positions. This replication is a natural consequence of implementing part-whole relationships with pairwise connections. A part-whole relationship involves three different things: The part, the whole, and the role that the part plays within the whole. In the conventional implementation using pointers, the role is

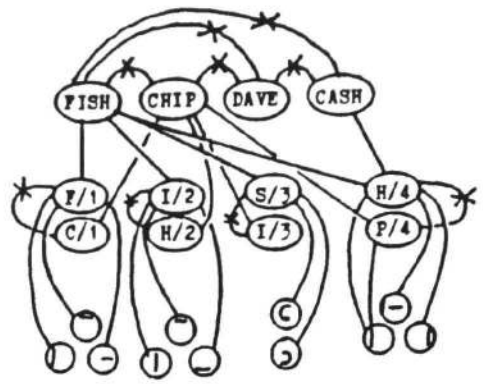


Figure 1: Part of a network used for recognizing words. Only a few of the units and connections are shown. The connections between alternative hypotheses at the same level are inhibitory.

encoded by which field the pointer is in. A pairwise connection between neuron-like units does not have anything equivalent to a field, and so the destination of the connection is used to represent both the field and the contents of the field. Thus, instead of having a single role-independent representation of H which is pointed to from many different fields, we have many different "role-specific" representations. Activity in any one of these units then represents the conjunction of an identity and a role.

At first sight, the fully-parallel implementation seems very wasteful because it replicates the apparatus for representing and recognizing letters across all the different roles. However, the replication has some useful consequences. It makes it possible to recognize different instances of the same letter in parallel without any of the contention that would occur if several different processes needed to access a single, central store of knowledge simultaneously. Also, when letters are used as cues for words, it is not just the letter identities that are important. It is the conjunction of the identity and the spatial role within the word that is the real cue. So it is very convenient to have units that explictly represent such conjunctions.

In addition to the expense of replicating the recognition apparatus across all roles, the fully-parallel network has several other problems :

1. The replication raises the question of how, if at all, the multiple different role-specific representations of a given letter are related to one another.

2. As we go down the hierarchy, there are less and less units available for representing each constituent.

3. In a network which is not fully connected, it is not at all obvious how new knowledge can be incorporated without growing new connections.

For relatively shallow, man-made hierarchies of the kind that are important in reading or speech recognition, it may be tolerable to always devote less units to representing smaller fragments of the overall structure. But for domains like normal visual scenes this strategy will not work. A room, for example, may contain a wall, and the wall may contain a picture, and the picture may depict a room. We need to be able to devote just as much apparatus to representing the depicted room as the real one. Moreover, the very same knowledge that is applied in recognizing the real room needs to be applied in recognizing the room in the picture. If this knowledge is in the form of connections and if the knowledge is not duplicated there must be a way of mapping the depicted room into an activity pattern on the very same set of units as are used for representing the real room.

The distributed version of method 1

The Rumelhart and McClelland network uses localist representations in which each entity is represented by activity in a single unit. Localist representations are efficient if a significant fraction of the possible entities are present on any one occasion or if the knowledge associated with each entity has little in common with the knowledge associated with other, alternative entities. Both these conditions hold quite well at the level of letter recognition. For the more natural part-whole hierarchies that occur in everyday scenes, neither condition holds. Only a tiny fraction of the possible objects are present on any one occasion, so if one unit is devoted to each possible object almost all the units will be inactive. This is a very inefficient way to use the representational capacity. Also, different objects, like a cup and a mug, may have similar appearances and may make similar predictions. This means that there can be a lot of useful sharing of units and connections. Most of what we know about cups and mugs could be associated with a unit that is active for either a cup or a mug. If this method of sharing is taken to its logical conclusion we arrive at distributed representations in which each object is represented by activity in many units and each unit is involved in the representation of many objects (Hinton, McClelland, and Rumelhart, 1986).

One major advantage of using descriptions rather than single units as representations is that it is possible to create representations of novel objects (and also novel role-specific representations) by using novel combinations of the same set of primitive descriptors. This avoids the problem of having to find a suitably connected unit for each novel object. However, the other two difficulties of the fully parallel method are not solved by simply using distributed representations.

Method 2: Sharing recognition apparatus within a level

Distributed representations provide a way of sharing units and connections between alternative objects or alternative role-specific representations. In this respect they work just like pointers in a conventional computer memory. Instead of using a separate bit for each possible object that could be pointed to, each bit is shared between many possible alternative objects. As a result, a word of memory can only point to one object at a time. [3] The following analysis of the functions performed by a role-specific representation suggests a quite different and complementary method of sharing which can be used to share connectionist apparatus between the different role-specific instances that occur within one whole. In the Rumelhart and McClelland model each role-specific letter unit has three functions:

1. It recognizes the occurence of that letter in that spatial role. The recognition is accomplished by having appropriately weighted connections coming from units at the feature level.

2. It contributes to the recognition of words. This is accomplished by its connections to units at the word level.

3. Its activity level stores the results of letter recognition.

[3] Some ancient implementations of LISP actually use two separate role-specific representations within one word so that the first part of a word can point to one object and the second part can point to another.

There is an alternative model which uses role-specific letter units for functions 2 and 3, but not for function 1. Instead, it uses a single letter-recognition module which is applied to one position within the word at a time. Once the letter at the current position has been recognized, the combination of its identity and its position within the word activates a role-specific letter unit which acts as a temporary memory for the results of the recognition and also contributes to the recognition of the word (see figure 2).

The letter-recognition module must be applied to one letter at a time and so there must be extra "attentional" apparatus which selects out one portion of the parallel input (which contains features of all the letters), maps this portion into the input of the letter-recognition module, and also creates an explicit representation of where the currently selected letter lies within the word. Actually, the Rumelhart and McClelland model presupposes that there is apparatus of a similar kind in order to pick out the features of one word within a sentence or to cope with changes in the position of a word. So the new model does not require any qualitatively new attentional apparatus, it just requires it at the level of letters instead of at the level of words.

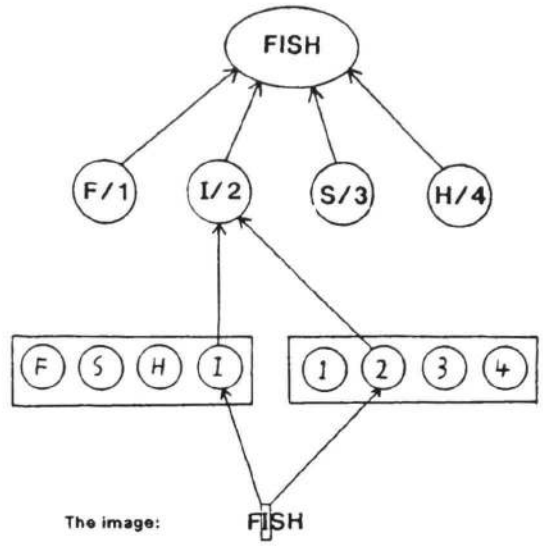


Figure 2: Some of the apparatus required to store the sequence of outputs of a single, sequential letter-recognition module in order to recognize a word. The network is "attending" to the second letter of the word. Notice that the role-specific units do not need to be able to recognize letters. The apparatus required for mapping the appropriate part of the input into the letter recognition module is not shown.

A connectionist dilemma

By sharing the recognition apparatus within a level, the network captures the regularity in the appearance of different instances of the same letter but it loses the ability to recognize all the letters of a word in parallel. This illustrates an important dilemma: If the knowledge is in the connections we can either capture the regularity by using a single module sequentially or we can have parallel recognition by using many modules at once. McClelland (1986) describes a clever but inefficient way out of this dilemma: There is a single central representation of the knowledge that is copied *during recognition* to produce the required parallel recognizers.

Figure 2 suggests an alternative way out. The serial recognizer could be used to train the parallel, role-specific recognizers. A network that time-shares a serial recognizer already requires role-specific units for storing its successive outputs. If these role-specific units had some connections to the perceptual input, they could learn to use these connections to predict the outcome of the serial recognition. The canonical, time-shared representation of the knowledge would then be acting as a superviser for the parallel recognition hardware. So the canonical knowledge would be transferred to the parallel recognition apparatus during learning which is much less demanding than transferring it during recognition. The hard part of learning is deciding what internal representations to use. Once this has been decided by the serial recognizer, it should be relatively easy to replicate this knowledge.

Method 3: Sharing across levels

There is one limitation of within-level sharing that is unimportant in the domain of reading but is very important in most other domains where the same knowledge can be applied at many different levels. For reading the knowledge is quite different at each level: Knowledge about the shape of a letter is quite different from knowledge about which sequences of letters make words, so there is little point in trying to use the same set of connections to encode both kinds of knowledge. In most natural domains, however, wholes and their parts have much in com-

mon. One example has already been given in which a room contains a picture that depicts a room. Another example is the sentence "Bill was annoyed that John disliked Mary." One of the constituents of this sentence "John disliked Mary" has a lot in common with the whole sentence. The same kind of knowledge is needed for understanding the constituent as is needed for understanding the whole. This is also typical of visual scenes which generally have just as much richness at every level.

If we consider how to map a part-whole hierarchy into a finite amount of parallel hardware there are three broad approaches:

1. The fully-parallel model uses a one-to-one mapping. Each object in the part-whole hierarchy is always mapped into the same set of units, and each set of units is always used to represent the same object.

2. Within-level sharing uses a many-to-one mapping. Many different objects at the same level can be mapped into the same set of units in the serial recognition apparatus. But whenever one of these objects is represented, it is represented in the same units.

3. Between-level sharing uses a many-to-many mapping. It allows many different objects at the same level to be mapped into the same set of units, but it also allows the same object to be mapped into different sets of units depending on the level at which attention is focussed.

The idea that the same type of object might be mapped to different sets of units is inherent in the idea of role-specific representations, but the idea that the very same instance can be represented in different ways depending of the focus of attention is a much more radical proposal. It is equivalent to viewing the hardware as a window that can be moved up and down (in discrete steps) over the part-whole hierarchy (see figure 3). One node in the hierarchy is chosen as the current whole and all of the units in the main network are then devoted to recognizing and representing this whole. Some units are used for describing the global properties of the whole, and others are used for role-specific descriptions of the major constituents of the whole. The entire pattern of activity will be called the "Gestalt" for the current whole.

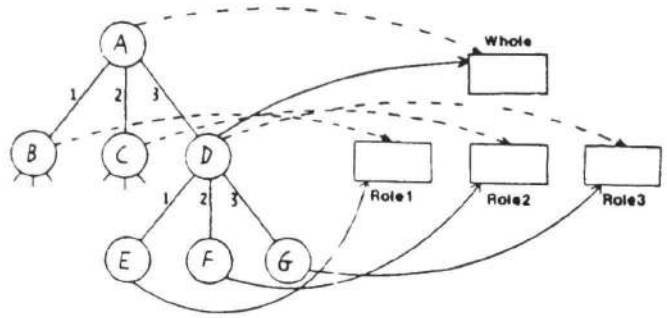


Figure 3: The solid and dashed lines show two different ways of mapping a part-whole hierarchy (on the left) into the same connectionist hardware (on the right). Notice that node D in the hierarchy can be represented by two totally different activity patterns that have nothing in common.

The crucial property of the moveable window scheme is that the pattern of activity that represents the current whole is totally different from the pattern of activity that represents the very same object when it is viewed as being a constituent of some other whole. In one case the representation occupies all of the main network and in the other case it is a role-specific description that occupies only the units devoted to that role.

The use of a many-to-many mapping raises many issues that do not arise or are not so important in the fixed mapping approach:

1. When the mapping between the world and the network is changed in such a way that one constituent of the previous whole becomes the new focus of attention, what kind of internal operations are required to convert the previous, role-specific description of that constituent into a full description that occupies the whole of the main network?

2. How is information about previous Gestalts stored so that the network can return to them later? The information cannot be stored as the activity pattern that the network settles to when the Gestalt is created because the very same network is needed for creating the next Gestalt.

3. How is the next mapping chosen?

There is not space here to address issue 3, but the following sections give a brief description of one way of handling issues 1 and 2.

Moving up and down the part-whole hierarchy

Figure 4 shows some of the extra apparatus that might be required to allow a connectionist network to move down the part-whole hierarchy by expanding a role-specific, reduced description into a full description of the role-filler. This corresponds to following a pointer in a conventional implementation. Notice that it is a slow and cumbersome process. Moving back up the hierarchy is more difficult. First, the full description of a part must be used to create the appropriate role-specific, reduced description of that part. This involves using the apparatus of figure 4 in the reverse direction. Then the role-specific, reduced description must be used to recreate the earlier full description of which it is a constituent. This requires some kind of content-addressable working memory for earlier Gestalts.

The obvious way to implement this working memory is to set aside a separate group of "working memory" units. If it is only necessary to remember one Gestalt at a time, this group can simply contain a copy of the pattern of activity in the network where Gestalts are formed. If several Gestalts need to be remembered at a time, several different groups could be used. Alternatively, a single group could be used provided that the various patterns of activity that need to be stored are first recoded in such a way that they can be superimposed without confusing them with one another. Examples of such encodings are described by Hinton (1981b) and Touretzky and Hinton (1985). Touretzky (1986) shows how this kind of working memory can be used to traverse and transform tree structures.

An interesting alternative implementation of working memory uses temporary modifications of the connection strengths in the network that is used for creating the Gestalt. Each internal connection in this network can be given two different weights: A long-term weight which changes relatively slowly and a short-term weight which is limited in magnitude, changes rapidly, and spontaneously decays towards zero. The effective connection strength at any time is simply the sum of the short-term and long-term weights. The long-term weights encode knowledge about which patterns of activity constitute good interpretations of the input to the network (i.e. familiar or plausible Gestalts). The short-term

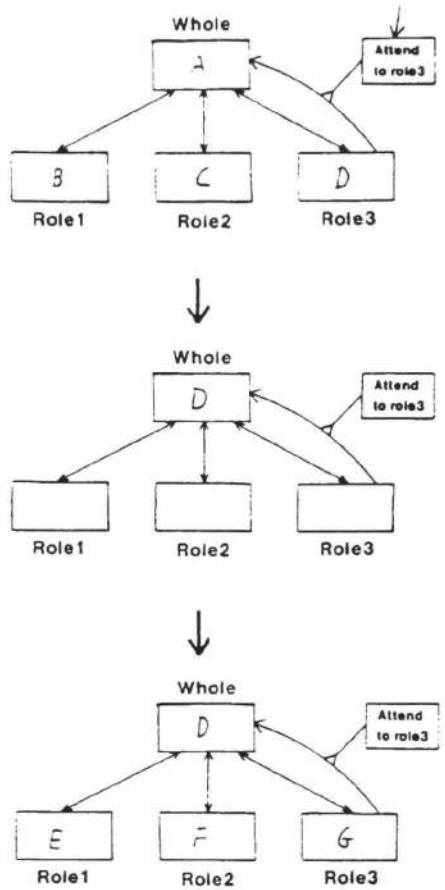


Figure 4: One way of using some additional hardware to allow the network to access the full description of node D from a role-specific reduced description. Even though the two descriptions correspond to quite different patterns of activity, their relationship should be non-arbitrary.

weights act as a contextual overlay [4] that encodes information about which patterns of activity occurred recently. If the network receives a rich external input which is incompatible with recently occurring Gestalts, it will settle to a new Gestalt and the short-term weights will act as noise (to which these networks are very resistant). If, however, parts of the external input are missing and the remainder fits some recently occuring Gestalt, the short-term weights will favor this Gestalt over other alternative Gestalts which would fit the partial input just

[4] Hinton and Plaut (1987) describe a very different use of this contextual overlay. It can be used to approximately cancel out recent changes in the long-term weights, thus allowing earlier memories to be "deblurred"

as well if the short-term weights were not considered. So the short-term weights will implement a content-addressable memory for recent Gestalts. Simulations (Hinton, 1973, unpublished) show that short-term weights can be used to allow the network to return to a partially completed higher-level procedure after executing a recursive call of the same procedure.

Conclusions

The combination of massively parallel constraint-satisfaction using reduced descriptions and relatively slow sequential access to full descriptions is a style of computation that is well-suited to networks of richly connected but rather slow processing elements. There is an inner loop of parallel, iterative processing in which the network performs a great deal of computation by settling into a state that satisfies constraints that are encoded in the connections. More elaborate computations which cannot be performed in a single settling are performed by a sequence of settlings, and after each settling the mapping between the world and the network may be changed. Changing the mapping corresponds to following a pointer (i.e. performing a remote access).

It is tempting to identify each change in the mapping between the world and the network with a single step in the network's "train of thought" This leads to an interesting view of what happens when a conscious cognitive process becomes automatic. Prolonged experience in a domain allows the network to develop reduced descriptions that make explicit the important regularities of the domain (see Hinton, 1986, for an example). This allows more of the computation to be done by interactions between the reduced descriptions, so there is less need to perform inherently sequential operations that change the way in which pieces of the task are mapped onto the parallel hardware.

References

W. D. Hillis. *The Connection Machine.* MIT Press, Cambridge, Mass., 1985.

G. E. Hinton. Learning distributed representations of concepts. In *Proc. Eighth Annual Conference of the Cognitive Science Society*, Amherst, Mass., 1986.

G. E. Hinton. Shape representation in parallel systems. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence, Vol 2*, Vancouver BC, Canada, 1981.

G. E. Hinton and K. J. Lang. Shape recognition and illusory conjunctions. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, Los Angeles, 1985.

G. E. Hinton, J. L. McClelland, and D. E. Rumelhart. Distributed representations. In D. E. Rumelhart, J. L. McClelland, and the PDP Research Group, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume I Foundations*, MIT Press, Cambridge, MA, 1986.

G. E. Hinton and D. C. Plaut. Using fast weights to deblur old memories. In *Proc. Ninth Annual Conference of the Cognitive Science Society*, Seattle, WA, 1987.

J. L. McClelland. The programmable blackboard model of reading. In J. L. McClelland, D. E. Rumelhart, and the PDP Research Group, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 2 Applications*, MIT Press, Cambridge, MA, 1986.

J. L. McClelland and D. E. Rumelhart. An interactive activation model of context effects in letter perception, part 1: an account of basic findings. *Psychological Review*, 88:375–407, 1981.

D. S. Touretzky. Reconciling connectionism with the recursive nature of stacks and trees. In *Proc. Eighth Annual Conference of the Cognitive Science Society*, Amherst, Mass., 1986.

D. S. Touretzky and G. E. Hinton. Symbols among the neurons: details of a connectionist inference architecture. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, Los Angeles, 1985.

Using Rules and Task Division to Augment Connectionist Learning

William L. Oliver and Walter Schneider

Learning Research and Development Center
University of Pittsburgh

Abstract

Learning as a function of task complexity was examined in human learning and two connectionist simulations. An example task involved learning to map basic input/output digital logic functions for six digital gates (AND OR, XOR and negated versions) with 2- or 6- inputs. Humans given instruction learned the task in about 300 trials and showed no effect of the number of inputs. Backpropagation learning in a network with 20 hidden units required 68,000 trials and scaled poorly, requiring 8 times as many trials to learn the 6-input gates as to learn the 2-input gates. A second simulation combined backpropagation with task division based upon rules humans use to perform the task. The combined approach improved the scaling of the problem, learning in 3,100 trials and requiring about 3 times as many trials to learn the 6-input gates as to learn the 2-input gates. Issues regarding scaling and augmenting connectionist learning with rule-based instruction are discussed.

Introduction

In this paper we compare human learning of a modestly complex task with connectionist learning that used the procedure known as "backpropagation" (Rumelhart, Hinton & Williams, 1986). We also consider a model that uses rules to divide the task into subtasks that can be separately learned with backpropagation. We examine the benefits of providing a connectionist system with a rule-based instructor that can reconfigure the system via attention to learn components of the task.

A critical issue for artificial intelligence and human learning involves finding learning algorithms that scale well. Learning time for an algorithm should not increase so dramatically with task complexity that it can only be applied to toy problems. Minsky and Papert (1988, p. 262) comment on the importance of the scale issue stating: "In the examination of theories of learning and problem solving, the study of such growths in cost is not merely one more aspect to be taken into account; it is the only aspect worth considering."

To the psychologist the problem of scale has critical importance because the time a biological system has to learn is limited. A learning algorithm that does not allow the organism to learn a task in its life time is of limited value.

Current connectionist algorithms may scale too poorly to account for human learning in many instances. Many tasks may be learned far more quickly by humans than by currently available connectionist procedures, because human learning can be guided by rules. Below we describe such a task in which humans required around 300 trials to learn. In contrast, currently our fastest learning simulations using only backpropogation required 68,000 trials. (see Figure 2 below). More importantly, human learning time did not increase with increases in the complexity of the task, whereas the learning times for the connectionist procedure significantly increased.

The study of connectionist learning is partially supported by an implicit assumption that humans provide an existence proof for simple, powerful learning algorithms that scale well. This assumption is likely to be false. By *simple* learning algorithms we mean algorithms that can map inputs to outputs by altering connection weights on each trial given the input and the desired output state of the system. This learning occurs without using explicit rules or focusing the network's attention on specific parts of the problem. Human learning in such situations is poor and does not scale well. Subjects take many trials to learn simple concepts involving very few feature dimensions (usually about 4) in psychological studies in which subjects are discouraged from forming verbal rules (e.g., Medin and Schaffer, 1978). Humans benefit greatly from focusing attention, instruction, hypothesis generation, and learning by imitation, none of which is present in traditional connectionist learning models. When learning a complex problem, such as family hierarchies (Hinton, 1985), a connectionist procedure must develop internal representations solely from the inputs and outputs that are specified on each learning trial. There is no mechanism to directly instruct the network about relationships among features (e.g., that *female* and *daughter* are correlated features such that daughters are always female). The backpropagation procedure can learn simple tasks of this sort, but learning often requires thousands of trials. We believe that both simple learning algorithms and rule-based learning will be necessary to account for human learning.

The human learning of chicken sexing (identifying young chicks as males or females) provides a contrast between learning by input-output mapping and learning by instruction on rules. Until recently chicken sexers had to learn their task on the basis of feedback from experts and on-the-job practice. It was claimed to have taken years for people to become proficient at this task (Biederman & Shiffrar, 1987). Biederman

and Shiffrar demonstrated that college students could perform a variant of the chicken sexing task as well as experts when provided with a classification rule. Only about a minute was needed to instruct the subjects on this rule, which focused subjects' attention on particular features and told them how to respond given the presence of those features. This example suggests that humans can learn complex relations via reinforced input-output mapping, but this learning method scales poorly and can be greatly improved by using attentional and instructional operations that are generally absent in connectionist learning.

We are examining connectionist architectures that include attentional focusing and instruction-based learning (Schneider & Detweiler,1987; Schneider & Mumme,1988; Schneider & Oliver, 1988). These architectures combine features from connectionist and production-system models. Rule-based processing allows an attentional mechanism to dynamically reconfigure connectionist networks so that critical features become salient and a task can be decomposed into subtasks of smaller scale. Using rules allows rapid initial learning of the components of the task and the serial execution of each component, as occurs in Anderson's (1983) ACT* or Laird, Rosenbloom and Newell's (1986) SOAR. Connectionist learning within the architecture can convert serial processing of the component rules to parallel processing as a consequence of practice. In addition, the mutual constraint nature of connectionist processing provides a best-match mapping of inputs to outputs that is less brittle than rule-based matching processes.

In this paper we examine the benefits of task decomposition by comparing the human learner to a connectionist learning system with and without task decomposition. We examine the effect of learning as a function of the complexity of the task. The task involved learning digital input-output mappings for six digital logic gates (AND, OR, XOR and the negated forms of the rules) for either 2, 4 or 6 inputs per gate. We have studied this task extensively in the acquisition of human troubleshooting skill (Carlson, Sullivan & Schneider, 1988a, 1988b). When learning this task, human subjects describe their processing as having three stages. The first stage is encoding the inputs as all 1's, all 0's, or mixed. The second stage is mapping the coded input and the gate type to the expected output of a 0 or 1. The third stage involves applying the negation operator when it is required to reverse the output. Subjects were instructed on rules for each stage and then required to learn 2-, 4-or 6- input gate problems. Connectionist learning without decomposition was examined in a network that mapped the inputs to the outputs through a single layer of hidden units. Input-output pairs were presented to the network, and backpropagation learning (Rumelhart et al., 1986) was used to modify the connection weights. Connectionist learning with decomposition was examined in a network composed of three modules, one for each stage. Each module had an input layer and an output layer. During training, each module received input and output information for each stage and propagated error only within its own stage.

Human Learning of Digital Logic

The computational properties of connectionist models have been studied by examining how they learn boolean functions (e.g., Minsky & Papert, 1988; Rumelhart et al., 1986; Volper & Hampson, 1986). Interestingly, research on digital trouble shooting has also looked at how subjects learn boolean logic in the laboratory (Brooke & Duncan, 1983; Carlson et al., 1988a, 1988b). In order to compare a connectionist model's learning with human learning, we designed an experiment that required subjects to learn several boolean functions and later had the model learn the same set of functions. We were mainly interested in whether increasing the complexity of the task by increasing the number of inputs to the functions would make the task much more difficult to learn.

The subjects in this experiment were University of Pittsburgh undergraduates with no experience in digital logic. A between-subjects experimental design was used; one group of 8 subjects learned digital logic gates with 2 inputs and another group of 9 subjects learned gates with 6 inputs. The subjects' task was to learn the rules for the gates to a high level of accuracy while responding as quickly as possible. Subjects typically reach an asymptotic accuracy of only about 92% in this task (Carlson, et al., 1988b). Their errors are random, suggesting causes other than rule learning (e.g., attention shifts, speed-accuracy tradeoffs) for the less-than-perfect performance.

The subjects learned six digital logic rules--AND, NAND, OR, NOR, XOR, XNOR. The subjects predicted the correct outputs when given different combinations of 0's and 1's as inputs for the various logic gates. The inputs to the gates were randomly determined with certain constraints on each trial (see below). The gates and their inputs appeared one at a time on a CRT screen, and the subjects indicated the correct output (0 or 1) by pressing labelled keys. A computer controlled the sequencing and presentation of the stimuli and gathered data on the accuracy and speed of the subjects' responses. Feedback on the correctness of response was provided after each trial. The subjects were given verbal rules during the early part of the experiment for each gate, such as the following rule for the AND gate: "if the the inputs are all 1's respond 1; if the inputs are mixed (0's and 1's) respond 0; and if the inputs are all 0's respond 0."

When a help key was pressed, the appropriate rules for a gate appeared in the upper-left-hand corner of the screen. An introduction to the three gate types (AND, OR, and XOR) involving 24 trials per gate was followed by 36 practice trials responding to gates and inputs selected at random. The subjects were then given instructions on how to carry out negation for the different gates (NAND, NOR, and XNOR) and given 24 trials of practice on each of these gate types. An additional 36 practice trials followed in which the negated gates were selected at random and presented to the subjects. In the final part of the experiment, the subjects responded to 300 gates selected at random from the entire set, including negated gates. The subjects could rest briefly after blocks of 50 trials, and use of the help key was not permitted.

In order to vary the complexity of the task, the number of inputs to the gates differed between groups of subjects. One group of subjects saw gates with 2 inputs and another group saw gates with 6 inputs. Because increasing the number of inputs dramatically changes the proportion of 1 and 0 responses for a given gate, a constraint was placed on the sampling of input combinations for the 6-input condition. For the 6-input gates, the probability of sampling certain input combinations (e.g., the all 1's case for the AND gate) was increased to maintain the same proportions of 0 and 1 responses as occurred in the 2-input condition. Without this constraint on the generation of input combinations, the subjects would be biased towards always giving the same response for a particular gate--for example, they would be biased towards responding 0 to every AND gate because the probability of that answer being correct would be .98.

Human Learning Results

The subjects responded correctly on a high proportion of trials (92%) during the final 300 trials of practice. The mean percentages of correct responses over 50-trial blocks were 89, 90, 94, 95, 94, and 93% for blocks 1 through 6 respectively. Hence, the subjects started this final part of the experiment with high accuracy and became somewhat more accurate with the additional practice. An analysis of variance that included the variables for input condition and 50-trial blocks indicated that there were significant differences in accuracy among the blocks, $F(5,75)=4.60$, $p<.001$. The main effect for input condition was not significant, $F(1,15)<1$, nor did input condition interact with blocks, $F(5,75)<1$. The mean accuracies were 92% for the 2-input condition and 93% for the 6-input condition.

An analysis of the subjects' response times also failed to show differences between the 2- and 6-input conditions. The subjects responded faster, on average, to the 6-input gates (2.18 seconds) than to the 2-input gates (2.31 seconds), but this difference was not significant, $F(1,15)<1$. As one might expect, there was a significant speed-up over blocks, $F(5,75)=14.52$, $p<.001$; the means for the eight 50-trial blocks, beginning with block 1, were 2.77, 2.34, 2.19, 2.10, 2.02, and 1.90 seconds. The variables input condition and 50-trial block did not significantly interact, $F(5,75)=1.15$, $p>.34$.

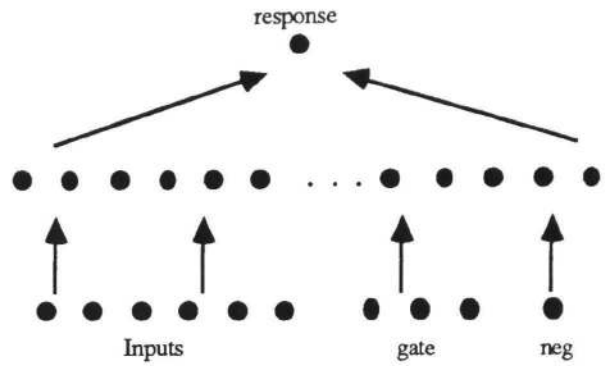


Figure 1. The configuration of the network that learned the 6-input gates without task division.

In summary, the initial 216 trials of training brought the subjects to a high level of accuracy. The final test blocks showed that the subjects could maintain, and even improve, this accuracy when they were tested on the different gates at random. There was no indication that the six input gates were more difficult to learn than the 2-input gates.

Connectionist Learning Without Task Division

We also examined connectionist learning of the digital logic task using the backpropagation learning procedure. A software package developed by McClelland and Rumelhart (1988) was used to model the task. To find out how changing the number of inputs would affect learning, we modelled learning of 2-, 4- and 6-input gates.

The networks trained with backpropagation were feed-forward networks having either 6, 8 or 10 units in the input layer. Each network had 20 hidden units, and a single output unit. The input layer consisted of 3 units to encode gate type, 1 unit to encode negation, and 2, 4 or 6 units to encode the inputs (0's or 1's) to the gates. Figure 1 illustrates the network's configuration for learning the gates with 6 inputs. Different codes were used for the AND (100), OR (010), and XOR (001) gates, and the negation unit was set to 1 to represent the negated gates (NAND, NOR, and XNOR) and otherwise set to 0. The initial weights for the network were set to random values that varied uniformly between -0.5 and 0.5. The momentum parameter was set to 0.9. We tried a number of different learning rate parameters, and the simulations we report below used the parameters that

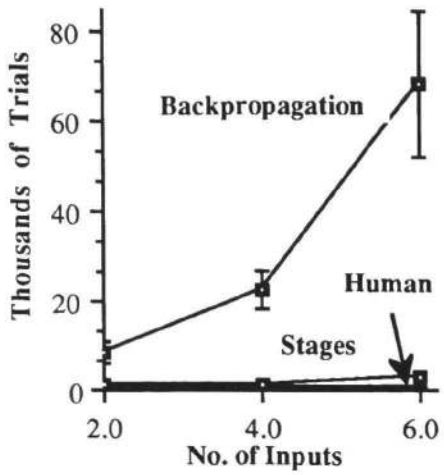


Figure 2. Trials to criterion for humans, backpropagation alone, and backpropagation with stages.

yielded the fastest learning. These learning rate parameters were .1, .07, and .02 for the 2-, 4-, and 6-input networks respectively.

Following the usual procedure for backpropagation, the networks were repeatedly presented with the complete set of patterns to be learned in cycles or "epochs." The networks were presented with patterns corresponding to all possible feature combinations for the gates and their inputs. Particular patterns in the 4- and 6-input simulations were repeatedly presented to the network within epochs to achieve the same proportion of 1 and 0 responses that subjects had encountered in the experiment described above. The weights were adjusted after each pattern so that the network learned over epochs to respond to the patterns with the appropriate 0's and 1's.

Each network's accuracy was tested at 10-epoch intervals during learning by presenting the set of training patterns to the network while learning was turned off. A network's response was assumed to be a 1 if the activation of the output exceeded .5, and 0 if its activation was less than .5 (possible activation values varied between 0 and 1). Ten simulations were run for the different network configurations, each starting with different random weights.

Figure 2 shows the number of trials (number of epochs times number of patterns per epoch) needed for each network to learn to the criterion of 100 percent accuracy. This criterion was used because the network's behavior was deterministic; if the network was less than perfect it would always err on the same patterns. These systematic errors, which are uncharacteristic of our subjects who performed above

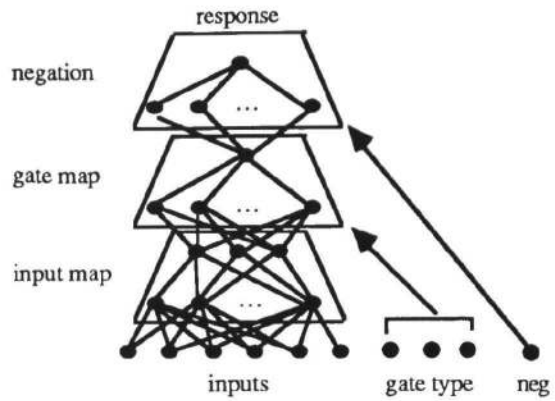


Figure 3. The configuration for the network that used task division to learn the 6-input gates.

90% accuracy, were taken to mean that the network had not yet learned the task.

As the complexity of the task increased, there was a substantial growth in the number of trials necessary to train the networks. Note that this growth contrasts dramatically with the lack of any complexity effect in the human data. This growth apparently resulted from the exponential increase in the number of patterns to be learned by the network; the number of patterns to be learned doubled with each additional input. There were 24, 96, and 384 patterns to be learned in the 2-, 4-, and 6-input conditions respectively. Generalization of learning among the patterns was insufficient to hold down the learning time.

Connectionist Learning with Task Division

Human learning may scale well in our task because of the subjects' abilities to divide the task into component tasks. These component tasks can be separately focused on during both instructions and performance of the task. The subjects' prior knowledge allows them to be instructed on the rules that apply to the component task and would, even in the absence of explicit instructions, allow them to form hypotheses about which feature combinations might be important. Such task division and use of prior knowledge are, of course, standard features in many simulations of cognitive processes, e.g., Anderson's ACT* (1983). Furthermore, the notion of information processing stages has played a fundamental role in cognitive psychology. Much reseach has been designed to identify stages of processing and discover how they interact (e.g., Sternberg, 1969).

To examine how task division might speed up learning in our task, we used backpropogation to learn the individual component tasks in a modular network. Figure 3 illustrates how the units that coded the gate inputs, gate type, and negation were used as inputs to the modules. The figure also shows

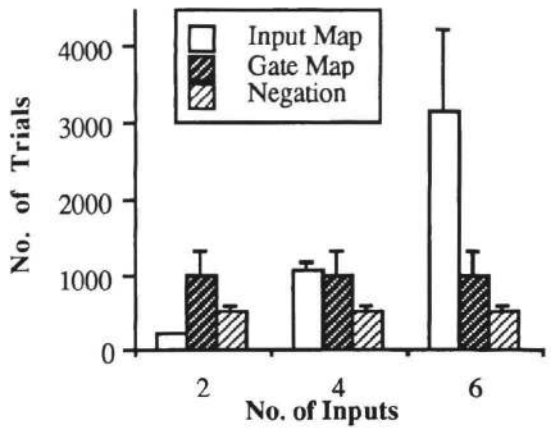


Figure 4. Trials to criterion as a function of subtasks and no. of inputs.

how the outputs from one module became the inputs to another module. The model had three modules, each containing a layer of input units, a layer of 10 hidden units, and a layer of output units. The first module (input map) was trained to recode 2, 4, or 6 inputs of 0's and 1's into codes representing either "all 0's", "all 1's", or "mixed." The second module (gate map) was trained to produce the correct responses (1 or 0) when given the recoded inputs and the codes for the gate types (AND, OR, XOR). The third module (negation) was trained to negate the output of the second module when negation was called for.

To assess total times for the model to learn the task, learning simulations were run for each module. Our results on learning times are based on 10 runs for each simulation. Each run was initialized to use a different set of random weights uniformly distributed between -.5 and +.5. For all modules, the momentum parameter was .9. The learning rate parameters for the input-map module were .5, .1, and .05 for the 2-, 4-, and 6-input conditions respectively. The learning rate parameter was .1 for the gate-map module and .5 for the negation module. These learning rate parameters were selected to enable rapid learning, but no major effort was taken to find the best parameters.

Figure 4 shows the mean number of trials needed to learn the component tasks for the different numbers of gate inputs. Figure 4 shows that recoding the input as 1's, 0's, and mixed requires substantially more trials as the number of inputs is increased. Assuming that learning can occur for all three modules during each trial, learning time would depend principally on the module that took the maximum number of trials to learn. This maximum value is plotted in Figure 2. It is clear from the figure that learning in this case scales considerably better than learning with back-propagation alone. It should be pointed out, however, that Figure 4 suggests that further increases in the number of inputs would require many more trials to learn if just three component tasks are assumed. If presented with even more inputs, the subjects would probably adopt additional coding processes to cope with increasing complexity, as is thought to occur when subjects chunk visual stimuli into familiar configurations (Bartram, 1978).

Discussion

We have examined human and connectionist learning of a modestly complex problem. The human subjects learned the task very quickly, reaching 90% accuracy by the second block of distributed practice. There was no evidence of any problem of scaling in the human learning data, with both the 2- and 6- input conditions reaching an asymptote of 93% in 358 trials. Reaction times declined substantially over trials, with the 2- and 6- input functions showing equivalent learning rates. In an extended study of human learning of digital gates (Carlson et al. 1988a) subjects took about 500 trials per gate or 3000 total trials to bring their response times below .8 seconds. When responding in .8 seconds, subjects have apparently shifted to a strategy of direct associative retrieval of the output of each stage given its input (see Carlson et al., 1988). To acquire this skill of automatic retrieval in the digital-logic task, subjects require about 5 hours of practice distributed over several sessions.

In sharp contrast to human learning, connectionist learning without task decomposition required about 68,000 trials to learn the 6-input case. Assuming that humans take about 6 seconds per trial, about 110 hours would be needed to perform 68,000 trials. This is far more than the 5 hours humans actually required. Even of greater concern than this long learning time, is the poor scaling shown in learning. The network required about 6 times as many trials to learn the 6-input rather than the 2-input case. The dramatic growth in the number of training trials suggests such a network could not learn an 8-input problem in the lifetime of a human.

Connectionist learning with task decomposition learned the 6-input case in about 3,200 trials and scaled fairly well, requiring 3 times as many trials than the 2-input case. The total number of trials compares reasonably well with the human performance, at least if we assume that the human connectionist processing is not well developed until humans can respond below 1 second. Connectionist learning with decomposition learned the 6-input case 21 times faster than without decomposition.

The above results suggest that combining rule-based and connectionist learning may provide the best of both types of computation. Initial rule-based

learning (as in ACT* and SOAR) can search a problem space and decompose a task into subtasks in reasonable amounts of time. Processing in this rule-based mode is slow, serial, and effortful as is a human novice during the controlled processing stage of skill acquisition (Shiffrin & Schneider 1977, Schneider & Detweiler 1987). Practice executing the rules allows connectionist learning to map the inputs to the outputs of each of the component tasks. The early rule-based processing decomposes a task so that smaller-scale tasks can be learned with connectionist procedures. This decomposition must identify the basic stages and the number of output states for each stage. Once tasks have been divided, connectionist learning need no longer perform gradient descent search in the power set of all possible connections, but rather has a more limited problem of mapping a small number of input states of each component task to a small number of output states for each component task. This use of task decomposition to make connectionist learning scale reasonably is an approach also advocated by Minsky (1988) to deal with the combinatoric explosion problem that occurs as task complexity increases.

Some readers might argue that our example provides an unfair test of connectionst learning and that our conclusions apply to only a limited set of tasks. We will briefly discuss four criticisms readers may have. *First, the problem chosen was a particularly difficult one for connectionist learning, since it included three levels of non-linearly separable problems (inputs, gates, negation).* We grant this, but it is a real task that humans have no difficulty performing if they are instructed. Learning combinatoric gates is still a toy problem and one that must be solved by any model of human learning. *Second, by instructing humans we gave away the answers.* We agree, but standard connectionist learning provides no mechanism for instruction. Since human learning can improve by many orders of magnitude with instruction, it is important to explore architectures that can benefit from instruction. *Third, different parameters or new learning algorithms may greatly speed learning in the present task, so that a connectionist procedure could learn the 6-input condition in a reasonable number of trials.* Perhaps, but the critical issue is whether new solutions will scale well. Task division and use of rules can always be used to reduce the scaling problem for any connectionist procedure, and it would be surprising if human learning would not make use of this property when learning new tasks. *Fourth, the present study shows that dividing tasks brings about faster learning, but there is no demonstration of how to implement the task decomposition in a parsimonious manner.* We are currently working on developing such an architecture.

We are developing a connectionist/control architecture (Schneider & Detweiler 1987, Schneider & Mumme 1988, Schneider & Oliver, 1988) that can implement rule-based learning and connectionist learning and that can benefit from instruction and task division. The architecture involves connectionist modules that transmit vector messages among modules. The control architecture uses an attentional gating mechanism that can modulate the transmission and reception of vectors among modules. Each module outputs information to the controller, indicating the degree of module activity and priority of its message. Controlled processing of the rules involves altering what messages are transmitted and compared in the network. For example, in digital gate learning, the rule would be of the form "if all the input module vectors match the lexical vector module (which contains a 1); then transmit the "ALL1s" code to the output of the input- coding module". Through changes in attentional gating, the network can be reconfigured to execute a process in as many stages as is required to perform the task. Intermediate states for each stage are represented not as specific units, but as random vectors.

Learning during the input coding stage illustrates how rule-based and connectionist learning interact in the connectionist/control architecture. The instructions to the model indicate that the input code must be encoded in one of three critical states and all the inputs map to these critical states. The network generates three random-state vectors and associates those to their respective rules (e.g., All1s = A; ALL0s=B, MIXED=C). The random vectors are similar to the *gensym* operator in LISP programs. During practice, the rule-based performance correctly solves the problem by serially executing the rules. On each trial the input and output of each stage are correctly set via the rule-based processing (Schneider & Mumme, 1988). Connectionist learning alters the connection weights to directly map the input to the output without the use of the rule. As opposed to doing a gradient descent search through the connection space for all possible output codes, the network needs only to learn how to map the input states to the instructed output states.

As the connectionist/control architecture learns a task, processing shifts from sequential, rule-based to association-based processing. Each module associatively maps its input to the output and this process cascades over a number of stages. This connectionist processing has two important advantages over rule-based processing. First, it is faster, because information is retrieved associatively. Second, it is not as brittle as rule-based processing because the mutual constraint match property of connectionist mapping will map the input to its closest matching output.

This may provide better generalization when the rule knowledge is ambiguous. The model follows the changes in human skilled performance as practice continues (Schneider & Detweiler 1987; Schneider & Mumme, 1988).

Summary

We have provided an illustration of the scaling problem exhibited by backpropagation when required to solve a modest complexity problem. We have shown that humans, if they are given instruction on the digital-logic task, show no effect of scale when the number of inputs to be learned was increased. The humans learned the most complex task 220 times faster (in terms of trials) than the connectionist simulation. We also evaluated a model using a task decomposition exhibited by the human subjects. Connectionist learning of the decomposed tasks scaled reasonably in this model, learning 21 times faster than the model without task decomposition for the 6-input case. We speculated that hybrid architectures provide a superior processing environment than either purely rule-based or connectionst processing environments. The hybrid architecture appears to scale well and learn at rates comparable to humans.

Acknowledgments

We thank Mark Detweiler for his comments on a draft of this paper. This research was supported by the Army Research Institute, under contract No. MDA903-86-C-0149 and Personnel and Training Research Programs Psychological Sciences Division, Office of Naval Research under Contract Nos. N-0014-86-K-0107 and N-00014-86-K-0678.

References

Anderson, J. R. (1983). *The architecture of cognition.* Cambridge, MA: Harvard University Press.

Bartram, D. J. (1978). Post-iconic visual storage: Chunking in the reproduction of briefly displayed visual patterns. *Cognitive Psychology, 10,* 324-355.

Biederman, I., & Shiffrar, M. M. (1987). Sexing day-old chicks: A case study and expert systems analysis of a difficult perceptual-learning task. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 13,* 640-645.

Brooke, J. B., & Duncan, K. D. (1983). Effects of prolonged practice on performance in a fault-location task. *Ergonomics, 26,*379-393.

Carlson, R. A., Sullivan, M. A., & Schneider, W. (1988a). *Fluency of component judgments in causal network problems.* Unpublished manuscript.

Carlson, R. A., Sullivan, M. A., & Schneider, W. (1988b). *Practice and working memory effects in building procedural skill for causal judgments.* Manuscript submitted for publication.

Hinton, G. E. (1986). Learning distributed representations of concepts. In *Proceedings of the Eighth Annual Conference of the Cognitive Science Society* (pp. 1-12). Hillsdale, NJ: Erlbaum.

Laird, J. E., Rosenbloom, P. S., & Newell, A. (1986). Chunking in Soar: The anatomy of a general learning mechanism. *Machine Learning, 1,* 11-46.

McClelland, J. L., & Rumelhart, D. E. (1988). *Explorations in Parallel Distributed Processing: A handbook of models, programs, and exercises.* In preparation.

Medin, D. L., & Schaffer, M. M. (1978). Context theory of classification learning. *Psychological Review, 85,* 207-238.

Minsky, M., & Papert, S. (1988). *Perceptrons (Expanded Edition).* Cambridge, MA: MIT Press.

Minsky, M. (1988). Preface: Connectionist models and their prospects. In D. Waltz and J. A. Feldman (Eds.), *Connectionist models and their implications: Readings from cognitive science* (pp. vii-xvi). Norwood, NJ: Ablex.

Rumelhart, D. E., Hinton, G. E., Williams, R. J. (1986). Learning internal representations by error propogation. In D. E. Rumelhart and J. L. McClelland (Eds.), *Parallel Distributed Processing* (pp. 318-364). Cambridge, MA: MIT Press.

Shiffrin, R. M., & Schneider, W. (1977). Controlled and automatic human information processing: II: Perceptual learning, automatic attending, and general theory. *Psychological Review, 84,* 127-190.

Schneider, W., & Detweiler, M. (1987). A connectionist/control architecture for working memory. In G. H. Bower (Ed.), *The Psychology of Learning and Motivation* (Vol 21, pp. 54-119). New York: Academic Press.

Schneider, W., & Mumme, D. (1986). *Attention, automaticity and the capturing of knowledge: A two-level architecture for cognition.* Unpublished manuscript.

Schneider, W., & Oliver, W. L. (1988). *An instructable connectionist/control architecture: Using rule-based instructions to accomplish connectionist learning in a human-time scale.* In preparation.

Sternberg, S. (1969). The discovery of processing stages: Extensions of Donder's method. *Acta Psychologica, 30,* 276-315.

Volper, D. J., & Hampson, S. E. (1986). Connectionist models of boolean category representation. *Biological Cybernetics,* 54, 393-406.

Analyzing a connectionist model as a system of soft rules

Clayton McMillan & Paul Smolensky

Department of Computer Science & Institute of Cognitive Science
University of Colorado, Boulder

With the rise to prominence of the connectionist or parallel distributed processing approach to cognitive modeling, the issue of the relation of such models to rule-based descriptions has been a consistent source of debate. It is our purpose in this paper to show that rather than regarding these approaches as completely mutually exclusive, there is insight to be gained in viewing standard connectionist models from a rule-based perspective. Our strategy is to show how a classic PDP model can be decomposed and viewed as a kind of rule-based model. We start with a summary of the PDP model we studied: Rumelhart and McClelland's (1986) model of acquisition of the past tense in English. This model is a natural choice, partly because it – and Rumelhart and McClelland's claims about its implications for the relation between connectionist models and rule-based accounts – has recently been the center of considerable controversy (Lachter & Bever, 1988; Pinker & Prince, 1988).

The past tense model

The past tense model simulates how children acquire the past tense of English verbs. It was designed to test the power of the PDP approach in what has long been considered the domain of rule-based models, natural language description and acquisition. Upon repeated presentation of verb stems and their corresponding past tense forms, the model learns a set of weights capable of producing the past tense of all 460 verbs in the corpus, plus many others not in the corpus. We are primarily concerned with the model after learning has been completed by presenting a corpus of verbs over 200 training cycles, and this final set of weights has been achieved.

The model is a bipartite graph. A phonetically spelled representation of the input – the stem of an English verb – is translated into a subset of 460 *Wickelfeatures*. These 460 Wickelfeatures are *position-independent, context-dependent phonetic features* and represent a fine-grained but somewhat restricted representation of phonemes present in the English language. For example, [*Back Vowel Front*] is a Wickelfeature present in any phonetic string that, somewhere, contains a vowel preceded by a back phoneme and followed by a front phoneme. This Wickelfeature is present in /kAm/ (*came*); there are 15 other Wickelfeatures present in this context-dependent representation of the vowel. Each Wickelfeature corresponds to one of 460 input units in the model. The input pattern is presented to the model by activating each of the input units corresponding to Wickelfeatures present in the input stem. Activation then passes once across the connections to a set of 460 output units, each of which also represents a Wickelfeature. As a result, units in the output become either on or off, depending upon the values of the weights on the connections, according to a certain stochastic rule. These output units indicate the Wickelfeatures present in the past tense form of the input verb.

Interpreting the output of the model is a bit complex. The degree to which the model prefers a given target output string over other possible strings is quantified in several ways. The most complex measure is *response strength*, which is computed by a rather complex network for decoding the output Wickelfeatures into output phoneme strings. A simpler measure Rumelhart and McClelland used is

$$1 - [\textit{fraction of target 1s not matched} + \textit{fraction of target 0s not matched}]$$

We will simply call this the *feature match* between the target and output. This can be defined for a single verb, or for a set of verbs; in the latter case, the two fractions appearing in the formula are each computed once over the whole set of verbs.

In general, if the correct response has a feature match in the range .50–.60, the decoding network tends to produce a response strength for the correct response that is greater than that of any other responses, but its superiority is often weak. With feature matches higher of about .65, the superiority in response strength starts to become pronounced.

The behavior of the model is regulated by the weights on the connections. The weights may be viewed as a matrix, where the weight on the connection between input unit i and output unit j occupies location (j, i) of the matrix: w_{ji}. We will refer to this matrix as $\mathbf{W}_{sim}$ because it is generated by training the model on an entire corpus of verbs simultaneously.

If this were a rule-based system the behavior would be regulated by rules that transform the verb stems into the past tense. After training, rules are *implicitly* present, although inaccessible, in $\mathbf{W}_{sim}$. It is this set of inaccessible rules that we wish to extract from $\mathbf{W}_{sim}$. Our strategy is to decompose $\mathbf{W}_{sim}$ into *several* weight matrices, each of which may be considered to correspond to a rule.

Goal of this research

Rule-based views of the formation of the English past tense have been developed by Bybee and Slobin (1982) and Hoard and Sloat (1973). Although these views are quite different, they share a common ground: Each verb is marked as belonging to a certain verb class. For each verb class there exists one or more rules that transform those verbs into the correct past tense form. The combination of these rules and markings represent a rule-based description of this cognitive task.

Our goal is to decompose $\mathbf{W}_{sim}$ into separate matrices, one for each class of verbs. We use the verb classification of Bybee and Slobin for our decomposition. They have identified eight irregular and three regular classes of verbs, each identified by shared morphological and phonological characteristics. We will therefore decompose $\mathbf{W}_{sim}$ into 11 separate matrices.

In order to view the weight matrices derived from $\mathbf{W}_{sim}$ as rules, each matrix must generate the correct past tense for verbs in its class. Each such matrix will be called a *soft rule matrix*. In order that these define something like a rule *system*, there must be a means of combining these 11 soft rule matrices into a single *composite* matrix, $\mathbf{W}_{com}$. We seek a $\mathbf{W}_{com}$ that performs the task at a level comparable to that of $\mathbf{W}_{sim}$.

Decomposing the weight matrix

We have taken a very simple approach to decomposing $\mathbf{W}_{sim}$. $\mathbf{W}_{sim}$ is generated by training the model on all verbs in all classes simultaneously. In a sense, it is a soft rule matrix that generates the past tense of all verbs in the corpus. Taking this same approach, we have chosen to generate soft rule matrices for each of the 11 classes of verbs by training the model separately on each class of verbs. The result of this approach is 11 soft rule matrices, one for each class.

The exclusive case

In the simplest technique, the training for a given verb class involves exclusively the verbs in that class; this training regime will therefore be called the *exclusive* case.

The behavior of the soft rule matrices in the exclusive case may be illustrated as follows. Bybee and Slobin's verb class 1 is the set of verbs that don't change in the past tense, such as *beat* or *cut*. Given any verb in class 1, $\mathbf{W}_1$ will generate the correct past tense for that verb. For a verb not in class 1, the behavior of the model is unspecified and unpredictable. Generally, the output for a verb outside class 1 will be a pattern containing features common to verbs in class 1. The same behavior is exhibited by $\mathbf{W}_\alpha$ for all classes $\alpha = 1, ..., 11$ in the exclusive case.

In this sense these soft rule matrices differ from traditional rules. With a traditional rule one would

expect a conditional clause to determine whether or not the rule should fire. If the rule fires, it will perform a deterministic transformation upon the state of the system. Either the transformation will be performed because the condition is met, or the state of the system will not change. By contrast, the soft rule matrices $\mathbf{W}_\alpha$ have no conditional element: they will always "fire" when presented with input. The output is only predictable when the soft rule matrix is applied to input that it is designed to accept.

The null case

It is possible to change the training regime so that the soft rule matrix corresponding to a given verb class corresponds more nearly to a traditional rule in that it fails to "fire" when presented with input outside the class – that is, for such input it produces *zero* output. Because of this null output specification, we call this training regime the *null* case. In the null case, the matrix $\mathbf{W}_\alpha$ is generated by training the past tense model on verbs in class α with the verb stems and correct past tense as the target, just as in the exclusive case. However, in addition, the model is presented with all verbs from the original training corpus that are *not* in class α, with *null* target patterns instead of the correct past tense.

To implement the null training regime we need to modify the delta learning rule used by Rumelhart and McClelland. In training $\mathbf{W}_\alpha$, the goal for verbs outside class α is that the *net input to each output unit j be zero.* Thus, if the input to the network represents a verb outside class α, let net_j be the net input to unit j: $net_j = \sum_i w_{ji} a_i$ (Here a_i is the activity of input unit i.) Let δ_j be 1 if $net_j < 0$, -1 if $net_j > 0$, and 0 if $net_j = 0$. Then we change the weight according to the usual delta rule: $\Delta w_{ji} = \delta_j a_i$ This will drive the weights towards the desired target of $net_j = 0$.[1] (Here, as throughout, the thresholds on the output units are replaced by weights to a hypothetical input unit that is always on; these weights are modified exactly like all other weights.)

Combining soft rule matrices into a single system

We have adopted two methods for combining the 11 soft rule matrices into a single composite matrix $\mathbf{W}_{com}$: *linear regression* and *straight summation.* In both methods we are assuming that there exists a linear relationship between the separate soft rule matrices and $\mathbf{W}_{com}$.

Linear regression

The idea behind the linear regression technique is to search for some set of weighting coefficients for the 11 soft rule matrices such that the weighted sum of these matrices will generate a single matrix that is close to $\mathbf{W}_{sim}$ and therefore can be expected to behave in a similar fashion. In effect we wish to minimize an *error* which is the mathematical difference between $\mathbf{W}_{com}$ and $\mathbf{W}_{sim}$.

In this analysis, $\mathbf{W}_{sim}$ is viewed as the dependent variable and the 11 soft rule matrices as independent variables. The result of the analysis is a set of 11 coefficients c_α; each coefficient is multiplied by the corresponding soft rule matrix $\mathbf{W}_\alpha$ and the results are summed together to produce $\mathbf{W}_{com}$: $\mathbf{W}_{com} = c_1\mathbf{W}_1 + \cdots + c_{11}\mathbf{W}_{11}$. The error we minimized is the usual sum-squared measure of the difference between $\mathbf{W}_{sim}$ and $\mathbf{W}_{com}$:

$$error = \sum_{i=1}^{460} \sum_{j=1}^{460} \left[w_{ij}^{sim} - w_{ij}^{com} \right]^2$$

where the w_{ij}^{sim} are the weights in $\mathbf{W}_{sim}$ and w_{ij}^{com} are the weights in $\mathbf{W}_{com}$. Minimizing this with respect to the coefficients $(c_1, c_2, \cdots, c_{11}) = \mathbf{c}$ that determine $\mathbf{W}_{com}$ leads to: $\mathbf{c} = \mathbf{X}^{-1}\,\mathbf{x}$, where $\mathbf{X}$ and $\mathbf{x}$ are respectively the 11×11 matrix and the 11-dimensional vector defined by:

$$X_{\alpha\beta} = \sum_i \sum_j w_{ji}^{\alpha} w_{ji}^{\beta} \qquad x_\alpha = \sum_i \sum_j w_{ji}^{\alpha} w_{ji}^{sim}$$

Here the w_{ji}^{α} are the weights in $\mathbf{W}_\alpha$.

Straight summation

The straight summation technique is much simpler than the linear regression technique: $\mathbf{W}_{com}$ is simply the unweighted sum of all 11 soft rule matrices $\mathbf{W}_{\alpha}$. The straight summation of the soft rule matrices is a special case of of the linear combination considered in the previous section: the case in which all coefficients c_{α} have value 1.

In the null case, there is a theoretical basis for expecting that straight summation will produce a $\mathbf{W}_{com}$ that does a reasonable job of combining the capabilities of the individual rules. Imagine an input verb from class β being processed by this $\mathbf{W}_{com}$. The net input to each output unit j is precisely the *sum* of the net inputs contributed by each separate $\mathbf{W}_{\alpha}$: $net_j = \sum_{\alpha=1}^{11} net_j^{\alpha}$. Now for each $\alpha \neq \beta$, by the definition of the null case, $net_j^{\alpha} = 0$. Thus, $net_j = net_j^{\beta}$: the net input to each output unit from $\mathbf{W}_{com}$ is the same as the net input from $\mathbf{W}_{\beta}$ alone; thus, the output of the network under $\mathbf{W}_{com}$ is the same as under $\mathbf{W}_{\beta}$. But, again by the definition of the null case, the output from the soft rule matrix $\mathbf{W}_{\beta}$ for a verb in class β is the correct past tense for that verb.

So, provided it actually is possible to use the null training regime to develop soft rule matrices that satisfy the null specifications, straight summation is a mathematically sound means of soft rule combination.

Null soft rule matrices combined with straight summation can be likened to a rule system in which each rule carries relatively equal weight and functions independently of other rules, and in which rules can be thought of as firing in parallel because the order in which they fire is unimportant. Such a rule system is said to be *free* (Lewis, 1987). Free rule systems probably provide the best analog to the type of rule system we are viewing the connectionist network as embodying.

Summary of the results

To test the hypothesis that $\mathbf{W}_{sim}$ may be viewed as a set of soft rule matrices that have been combined to form a matrix $\mathbf{W}_{com}$, we generated the 11 soft rule matrices $\mathbf{W}_{\alpha}$ for both the exclusive and the null cases, and examined both the linear regression and straight summation techniques for combining them.

To better understand the relationship between the various soft rule matrices we also charted the development during training of the c_{α} coefficients in the linear regression technique. Each c_{α} may be viewed as indicating the weight of the contribution of $\mathbf{W}_{\alpha}$ to the composite matrix $\mathbf{W}_{com}$.

During training of the soft rule matrices we would expect certain soft rule matrices to emerge dominant over others. In particular, we would expect soft rule matrices representing the larger body of regular verbs to dominate soft rule matrices representing the smaller body of irregular verbs. This can be seen in Figure 1. In the first ten training cycles, we trained only on the same ten verbs that Rumelhart and McClelland used for these cycles (those they identified as the ten most frequent verbs in the Kucera and Francis (1967) corpus). Of these ten verbs, eight are irregular; in the first ten cycles, the ratio of the average irregular c_{α} to the average regular c_{α} is very large. As a large body of regular verbs is introduced to the model at cycle 11, this ratio virtually flips, quickly stabilizing at .37 in the exclusive case and 1.01 in the null case.

This reveals three important points: (1) the relative contribution of each soft rule matrix $\mathbf{W}_{\alpha}$ to $\mathbf{W}_{com}$ is established early in training, (2) in the exclusive case there is a substantial imbalance between irregular and regular soft rule matrices, and (3) in the null case the ratio is almost 1 to 1. Thus we might expect a better performance in combining soft rule matrices in the null case than in the exclusive case, because the various soft rule matrices are more nearly alike.

In order to measure the quality of performance of our "rule system," we tested the model separately

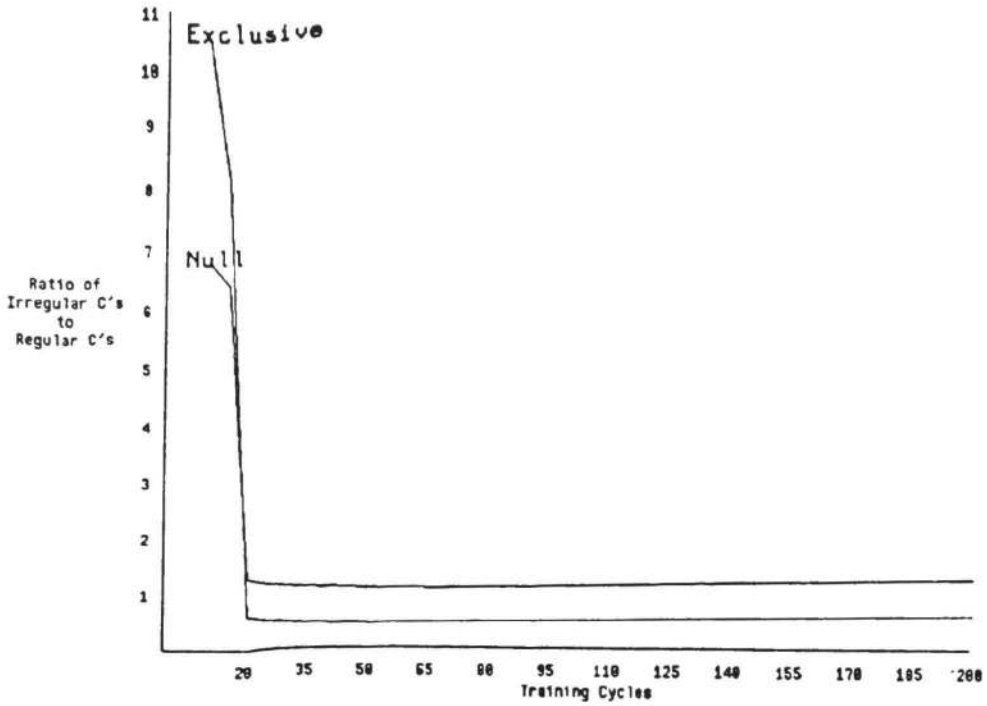


Figure 1: Ratio of average coefficients for regular verbs to average for irregular verbs.

on $\mathbf{W}_{sim}$, on $\mathbf{W}_{com}$, and on each of the soft rule matrices $\mathbf{W}_{\alpha}$. Our measure was the feature match (as defined above) over all the relevant verbs in the training corpus (the entire set for $\mathbf{W}_{sim}$ and $\mathbf{W}_{com}$; the verbs in class α for $\mathbf{W}_{\alpha}$).

We first consider the results in the exclusive case. As shown in Table 1, the average feature match of the soft rule matrices is .97; this is comparable to the feature match of $\mathbf{W}_{sim}$, .95. We may conclude that the soft rule matrices perform on average at the same level as $\mathbf{W}_{sim}$ when presented with words in the corresponding class.

Table 1: Performance (feature match)

Training technique	Original matrix $\mathbf{W}_{sim}$	Average of soft rule matrices $\{\mathbf{W}_{\alpha}\}_{\alpha=1}^{11}$	Composite matrix $\mathbf{W}_{com}$	
			Linear regression	Straight summation
Exclusive	.95	.97	.62	.54
Null	.95	.98	.52	.87

The quality of the two composite matrices in the exclusive case is indicated by a feature match of .54 and .62 for the linear regression and straight summation respectively. Substantially lower than $\mathbf{W}_{sim}$, these figures definitely are on the lower bounds of acceptability, as discussed earlier.

Why is the performance of $\mathbf{W}_{com}$ rather poor when the linear regression produces a matrix that minimizes the difference between its weights and the weights in $\mathbf{W}_{sim}$? Our explanation is as follows: There are an infinite number of matrices that perform the correct input/output transformation on the corpus of verbs across all 11 verb classes. The past tense model employs a method that simply produces *one* such matrix, $\mathbf{W}_{sim}$. Many of the other matrices will have a large difference from $\mathbf{W}_{sim}$, while nonetheless performing identically to $\mathbf{W}_{sim}$. Thus producing a matrix $\mathbf{W}_{com}$ in which there is minimal

difference between the weights in that matrix and $\mathbf{W}_{sim}$ does not guarantee that $\mathbf{W}_{com}$ is the best combination of $\mathbf{W}_{\alpha}$ for approximating the performance of $\mathbf{W}_{sim}$ on the training corpus.[2]

It is perhaps less surprising that the straight summation performs poorly in the exclusive case. While there is a theoretical basis for the soundness of the straight summation combination in the null case, there is none for the exclusive case.

The results of combining the soft rule matrices in the null case are also summarized in Table 1. Again we see that the performance of the individual soft rule matrices is comparable to that of $\mathbf{W}_{sim}$. The performance of $\mathbf{W}_{com}$ generated through linear regression is .52. The same explanation for the rather poor performance of $\mathbf{W}_{com}$ in the exclusive case applies here in the null case.

As predicted by our earlier analysis, the performance of the $\mathbf{W}_{com}$ generated through straight summation of the soft rule matrices is much better: The feature match is .87 for $\mathbf{W}_{com}$ when presented with the full corpus of verbs. In general, with a feature match of .87, we can expect the model's decoding network to consistently generate the correct past tense with very few, and weak, alternatives – if any alternatives are generated at all.

From this result we may conclude that the soft rule matrices in the null case may, in a sense, be viewed as free rules that may be applied separately or combined through straight summation into a single system.

Conclusion

We have shown how a classic PDP model, Rumelhart and McClelland's past tense model, may be decomposed into a set of "soft rule matrices." These rules may be applied separately or combined into a single system. Using the best technique, soft rule matrices trained in the null regime combined using straight summation, we can view the knowledge in this model's weight matrix in four approximately equivalent ways:

(1) Knowledge = $\mathbf{W}_{sim}$: The past tense model is a PDP model consisting of no rules.
(2) Knowledge = $\sum_{\alpha}\mathbf{W}_{\alpha}$: The past tense model is a PDP model in which the knowledge in the weights is a system built (by simple summation) of 11 individual matrices each handling a different subset of the input space.
(3) Knowledge = $\{\mathbf{W}_{\alpha}\}_{\alpha=1}^{11}$: The past tense model is a set of 11 separate noninteracting rules. Each rule is implemented as a PDP network.
(4) Knowledge = $\mathbf{W}_{com} = \sum_{\alpha}\mathbf{W}_{\alpha}$: The past tense model is a rule system combining 11 rules into one single system. The rules, and the way they combine, are defined via connectionist networks. The rules apply independently, in parallel.

In this work we have been exploring the hypothesis that the higher-level perspective provided by rule systems can help us understand the knowledge contained in a PDP network. Another hypothesis worth exploring is that PDP-based "soft" rules of the sort we have been considering might help, in simple domains, to alleviate some of the brittleness that has often plagued systems based on hard rules.

Our explorations clearly constitute the barest beginings. Especially important extensions of our work are to systems with hidden units and to methods for finding rule-decompositions of the sort we used here *automatically* – without the need for a prior (non-connectionist) analysis of the task (provided in our case by Bybee and Slobin). Nonetheless, we are encouraged that our preliminary foray has helped us understand the knowledge contained in a rather inscrutable weight matrix of well over 200,000 weights. We expect further useful results to come from explorations of how conceptual and technical tools of the PDP and rule-based frameworks can be used to strengthen each other.

Acknowledgements

This work has been supported by NSF grants IRI-8609599 and ECE-8617947 to the second author, by a grant to the second author from the Sloan Foundation's computational neuroscience program, and by the Department of Computer Science and Institute of Cognitive Science at the University of Colorado at Boulder.

We thank Dave Rumelhart and Jay McClelland for providing us with the simultation software for their model.

Footnotes

1. In practice, in order to avoid oscillations we set $\delta_j = 0$ if net_j is within a certain tolerance γ of the target value 0. If γ is chosen too small the system will generally fail to converge; on the other hand, $\gamma = \infty$ reduces the null case to the exclusive case. We found that $\gamma = 850$ was an acceptable value; this may seem large, but since there are 460 weights to each output unit, each with an integer weight and many of these greater than 1, the value of 850 is not large compared to a typical net input to an output unit.

2. Recall that our measure of difference is purely based on the weights in the matrices, and not in terms of their responses to the particular inputs in the training or testing sets.

References

Bybee, J. & Slobin, D. (1982). Rules and schemas in the development and use of the English past tense. *Language*, **58**, 265–289.

Hoard, J. & Sloat, C. (1973). English irregular verbs. *Language*, **49**, 107–120.

Kucera, H. & Francis, N. (1967). *Computational analysis of present-day American English.* Providence, Rhode Island: Brown University Press.

Lachter, J. & Bever, T. G. (1988). The relation between linguistic structure and associative theories of language learning—A constructive critique of some connectionist learning models. *Cognition*, **28**, 195–247.

Lewis, C. (1987). Composition of productions. In Klahr, D., Langley, P., & Neches, R. (Eds.), *Production system models of learning and development*, 329–359. Cambridge, MA: MIT Press.

Pinker, S. & Prince, A. (1988). On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. *Cognition*, **28**, 73–193.

Rumelhart, D. E. & McClelland, J. L. (1986). On learning the past tenses of English verbs. 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*, 217–271. Cambridge, MA: MIT Press/Bradford Books.

HOW TO SUMMARIZE THICK TEXT
(And Represent It Too)

Richard Alterman
Lawrence Bookman
Computer Science Department
Brandeis University

TEXT & SUMMARIZATION

1. Introduction

Consider the two pieces of text shown in figures 1 and 2. In one case we have a piece of text that is almost a caricature of a story, in the other case we have a piece of *thick text* taken from a book of folktales ("The clever peasant and the czars general:" Protter, 1961). The second story is thick because of the richness and complexity of the relationships amongst the events described; it is thick because it shows in detail the events of the story, rather than telling us their skeletal structure. One might argue, for example, that stories like the one described in figure 1 are unnatural because they are *pointless* (see Wilensky 1980,1982), or because their language is overly simplified. But, in contrast to "The Clever Peasant and the Czar's General," the reason for their unnaturalness becomes apparent: The first one is already a summary. In this paper we will describe a summarizer called SSs (**Summarization** Summarization Summarization...). that takes a thick piece of text, like the one shown in figure 2, and produces a skeletal piece of text, like the one shown in figure 1. SSs bases its summary on an event concept coherence analysis (ECC) (Alterman 1985,1988) of the text, as produced by a program called NEXUS.

Figure 3 depicts at the system level the relation between NEXUS and SSs. NEXUS produces an ECC representation by matching case encoded text against an associative network of event/state concepts. The ECC representation essentially sorts out and identifies the event concepts invoked by the text. SSs takes as input an ECC representation of a piece of text. It has basically two summarization strategies. The first technique works by first delineating boundaries of cohering event concepts and then extracting their core concept(s) - a shift in granularity The second technique works by identifying the major narrative thread of the story - a kind of outline. In this paper we present the first of these techniques. (See Alterman & Bookman, for details on the second of these techniques).

John was sailing his boat. Suddenly a gust of wind caught the sails. The boat capsized. John was very upset.

FIGURE 1. THE SAILING STORY

"Well, little farmer, what would you like?" the czar asked (1) when the peasant was brought (2) before him. "Nothing, father czar. I have come (3) only to bring (4) you a gift." And he opened (5) the lid of the metal chest. "And what would you like in return (6) for this gift of gold?" the czar inquired (7). "Father czar, just give (8) me a hundred lashes of the whip." "What?" the astonished czar exclaimed (9). "You ask (10) for a hundred lashes? You have brought (11) me gold and you want (12) a hundred lashes? No, no, no; you certainly deserve better (13)." "Please, father czar," the peasant insisted (14). "I don't want (15) any other reward. Just give (16) me a hundred lashes." So the czar reluctantly summoned (18) one of his

guards to fetch the whip. "Are you ready?" demanded (19) the puzzled ruler. "No, we must wait (20). I have (21) a partner who should share (22) the reward." A partner?" echoed (23) the bewildered czar. "Yes," the peasant answered (24). "When I came (25) to your door, the general would not let (26) me through until I vowed (27) that he would get (28) half my reward. So go ahead and start with him. Give (29) him the first fifty lashes."

FIGURE 2. A PORTION OF THE CLEVER PEASANT AND THE CZAR'S GENERAL

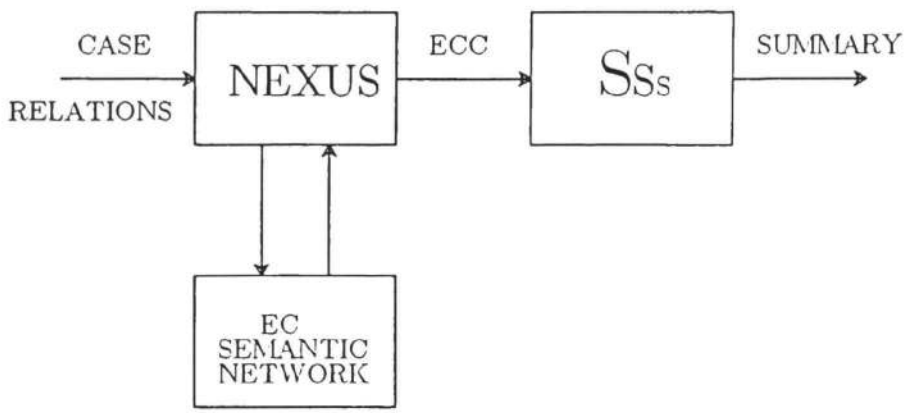


FIGURE 3. SYSTEM ARCHITECTURE

2. Event Concept Coherence

The ECC representation of the event descriptions of the text is a reflection of the relationships of the event/state concepts invoked by the text: the form of the knowledge and form of the text representation mirror one another. Its construction makes the following assumption: two event descriptions in a piece of text are *event concept coherent* if the positions of the concepts they invoke are proximal to one another in the underlying conceptual network. For NEXUS, knowledge about the relationships between event/state concepts are represented by an associative network, and an ECC interpretation captures the connectivity of a piece of text by essentially copying out and instantiating the relevant portion of the network. The representation was constructed so as to characterize the dependencies amongst the events in a causally neutral, but causally relevant form (Alterman 1988).

NEXUS uses intersection search as a basis for computing the representation (Alterman 1985; c.f. Quillian 1968; Charniak 1983; Norvig 1987). The input to NEXUS is the text in case notation form. NEXUS' program works in two stages. In stage one, it does a bi-directional breadth-first search to find a path between the event concepts previously introduced in the text (either explicitly or implicitly) and the new input event concept. In stage two, if a path is found, NEXUS propagates the case constraints along the path, simultaneously checking path consistency and performing some reference resolution. During the second stage default values are introduced to aid this process. The case constraints are also used by NEXUS to track the location of characters and to insure the correct sequencing of events in time.

Even though the representation of the text and the underlying conceptual network share the same structure, throughout this paper when we talk about the representation of the text, we will

emphasize the fact that the representation produced by NEXUS - because the representation it produces corresponds to the narrative structure of the text - is composed of multiple interconnected event concept trees (see Figure 4). A *concept tree* is recursively composed of all its subclass, part, and before/after descendants of a given event concept. Concept trees delineate the boundaries of individual event concepts and are temporally linked to one another forming what can be described as a *narrative stream*. The ECC representation characterizes the complex relationships amongst the events in a thick piece of text by grouping conceptually related events into tree-like structures for individual concepts, and temporal sequences of events for the narration as a whole.

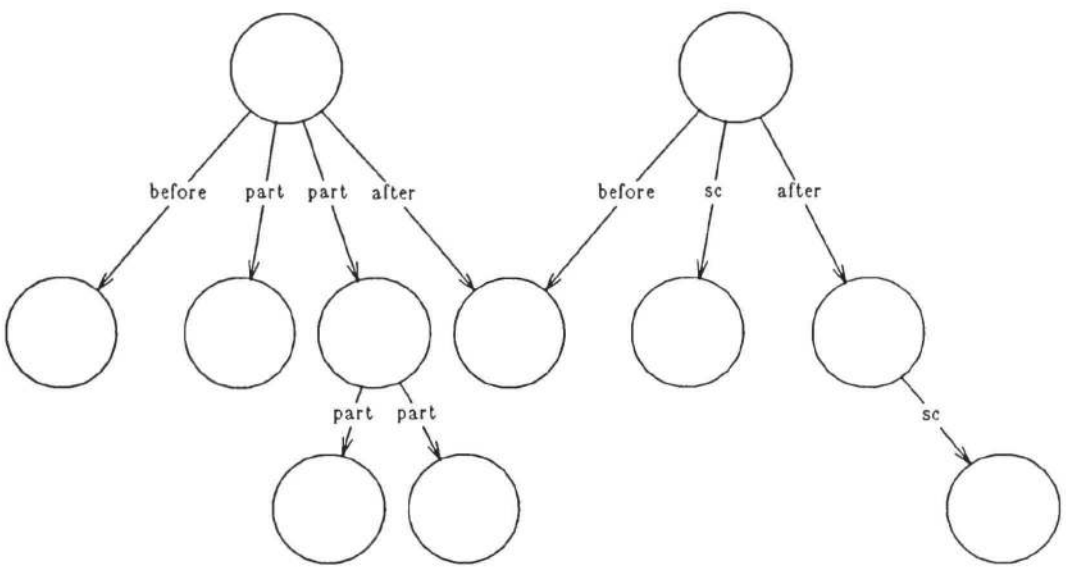


FIGURE 4. TWO INTERCONNECTED CONCEPT TREES.[1]

3. Thinning Techniques

Thick text is composed of a number of embellished text descriptions that are enmeshed together. A change in textual granularity from thick to thin text is achieved by systematically going through the text to identify and bundle together little pieces of text that conceptually form a larger event and then describing these concepts by their core concept. The shift in granularity is accomplished by a process of *delineation* and *extraction*. Delineation untangles the mesh of concepts by delineating the boundaries of the larger events and from these larger events the core concept is extracted.

Delineation occurs as a by-product of the ECC representation, as each concept tree in the ECC representation implicitly represents a delineated concept. Because of the hierarchical organization of the concept tree, extracting the central concept is relatively straightforward.

[1] There are actually seven coherence relations, since the *part*, *before*, and *after* relations used are further discriminated into (*subseq* & *coord*), (*ante* & *prec*), (*seq* & *consq*), respectively.

With one exception single concept trees can be summarized by extracting the top node in the concept tree. The basic idea of delineation and extraction is to consider each of the events in the representation produced by NEXUS and determine if it is the top of some concept tree. If it is a top, it is included in the summary, else it is discarded. Internally SSs represents each of the relations of the concept tree with a 3-tuple of the following form:

(Coherence_Relation Event/State1 Event/State2)

A *top* can be defined as follows:

For all events x,y and coherence relations c,
TOP(x) iff there exists no tuple such that (c,y,x) is true.

The exception occurs because NEXUS uses property inheritance to represent knowledge in the network. In a class/subclass relationship the subclass concept is preferred because it is more informative - the reader can predict all its relationships as well as the relationships it inherits from its ancestors.

4. An Example

Consider the portion of text from "The Clever Peasant and the Czar's General" shown in figure 2. When NEXUS and SSs were applied to this text using the delineation and extraction techniques it produced the following summary:

> The peasant had a audience with the czar. The peasant exchanged the chest of gold with the czar for a whipping. The peasant had a deal with a partner about sharing half the reward with him. The guards was to give the general a whipping. The czar requested his guards to fetch the whip.[2]

Figure 6 shows the analysis produced by NEXUS in which it identifies five event concept trees: *audience*, *exchange-gifts*, *request-of-underling*, *deal*, and *whipping*. Below is shown a description of some of the details of that analysis along with the correct sequencing of the concept trees.[3]

1. **The peasant has an audience with the czar**: beginning with event (2) the text introduces the fact that the peasant was brought before the czar. There is a conversation between the czar and the peasant (1,7,10,14,24).
2. **There is an exchange of gifts**: the peasant opens (5) a metal chest and the czar inquires what would he like in return (6). He tells the czar to give (8) him a hundred lashes.
3. **A request of underling**: the czar summoned (17) one of his guards to fetch (18) the whip.
4. **The peasant has made a deal with the general**: before he is to receive his reward he tells the general I have (21) a partner who should share (22) the reward.
5. **Administering the punishment**: the general was to be given (29) the first fifty lashes.

---

[2] Although we have some heuristics to handle the sequencing of events, these heuristics are not foolproof, as the event *request-of-underling* is not in the correct order

[3] Note the numbers in parenthesis indicate the events in the order in which they appear in the text in figure 2.

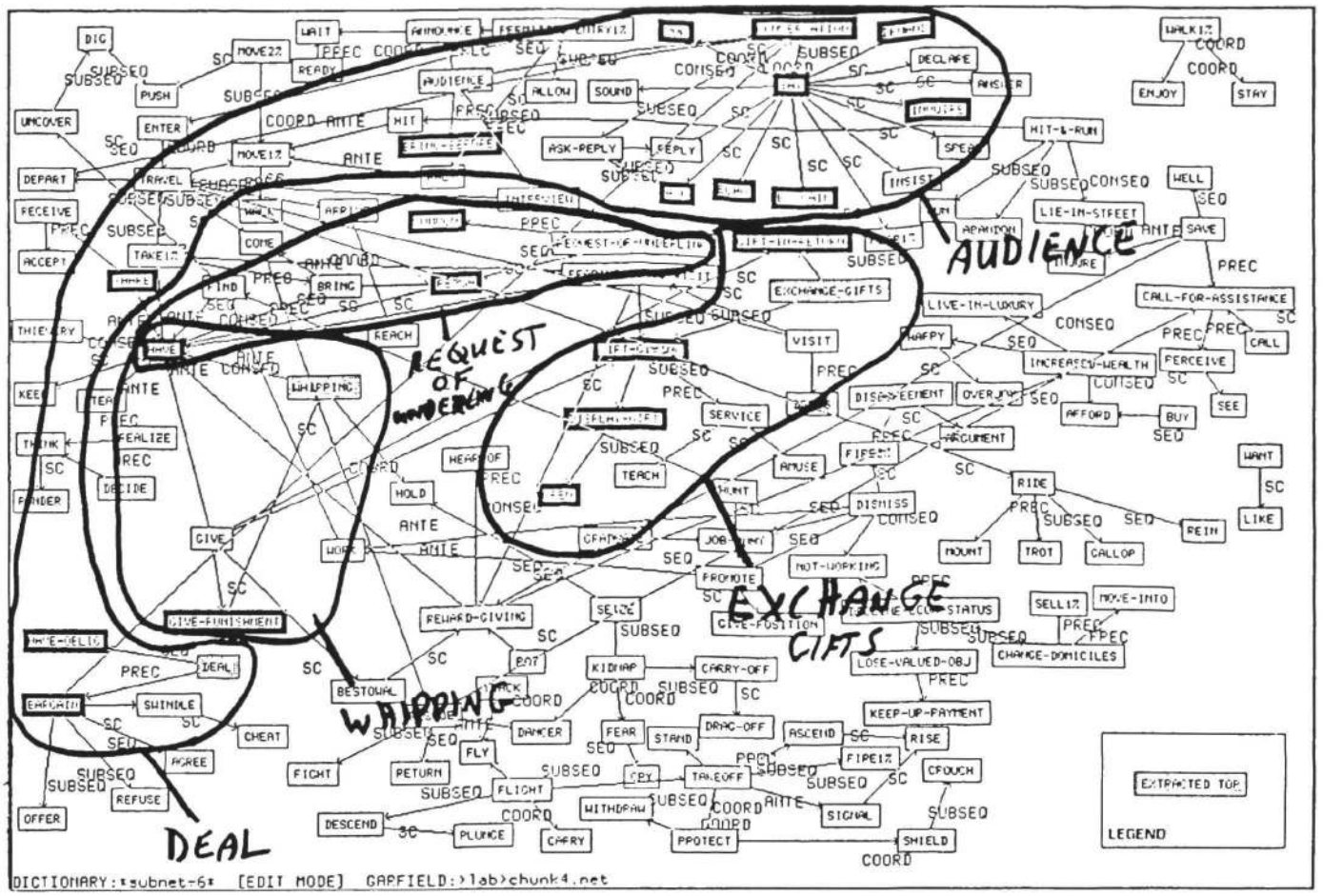


FIGURE 6. SCHEMATA OF "THE CLEVER PEASANT AND THE CZAR'S GENERAL"[4]

In terms of the narration of the text as a whole, the text is composed of three *chunks* of concept trees. (By chunks we mean interconnected concept trees.) In the first chunk, there is a connecting path between the concept trees *audience* and *deal*: Because the general and the peasant made a deal, the peasant has an obligation to the general, and this obligation is reported to the czar in the course of their conversation during the peasant's audience. In the second chunk there is a connecting path between the concept trees *request-of-underling* and *whipping*: The czar requests his guards to fetch the whip. As a result they have the whip in which to punish the peasant's partner, the general. The third chunk is the *exchange-gifts* concept tree which is not connected to either of the other chunks. SSs takes this underlying event structure and thins the text by extracting the tops of each of the concept tree within each of the interconnected chunks.

5. The Computational Literature on Summarization

Lehnert & Loiselle (1988) developed a scheme for summarizing text based on plot units (Lehnert, 1981). Plot units represent affect-state patterns. Lehnert identifies a number of primitive plot units (e.g. motivation, success, perseverance) which can be combined into more complex

[4] Note the shaded rectangles represent the ECC interpretation produced by NEXUS For further details of this analysis, see Alterman & Bookman (forthcoming) The graphics for this picture was created using AINET-2 (Chun 1986)

plot units (e.g. fortuitous problem resolution, fleeting success, giving up). Narrative text is represented by interconnected plot units and summaries are based on the identification of pivotal plot units, i.e., the plot units which are maximally connected.

Wilensky introduced a theory of summarization based on the identification of *story points* (Wilensky, 1982, 1980). Story points roughly correspond to the essential tension points of a story, i.e., what the story is about. The idea is that points represent what is interesting in a story and therefore likely to be included in a summary. Wilensky suggests some rules for recognizing points. The rules are based on his theory of goal interaction (Wilensky, 1983). If a character plans to go outside to get the newspaper and discovers it is raining outdoors, a *goal conflict* occurs between the goal to get a newspaper and the goal to stay dry. Wilensky argues that situations where goal interactions occur are potentially dramatic and consequently likely candidates as story points.

Both Wilensky and Lehnert are describing summarization techniques that attempt to identify critical points of interest. Both their techniques are based on analyses that impose high-level concepts in a top-down manner on the text. The difficulty is that there is a gap between the initial representation of the text and the actual representation used for computing both the story points and plot units respectively. An ECC representation for thick text, however, acts as a way station that sorts out some of the underlying event structure. It provides a schemata from which, perhaps these other summarizers can further reduce the volume of text by deciding what is of interest.

Research on story trees characterize the text by a set of meta-descriptions, e.g., episode or setting. Rumelhart (1975; Simmons & Correira, 1980; and Correira, 1980) summarized text based on a hierarchical organization of the text. Again we have a gap between a high-level structure and the events depicted in the text, which can be reduced by the analysis we are proposing. The work of Van Dijk (Macro-structures, 1976) suggests techniques for thinning out text, but organizes it in terms of a single hierarchical structure. The work described in this paper differs from the work on macro-structures with regards to the problems of thick text in two ways. First it emphasizes the role of events in the thinning process. Second the representation it produces does not take the form of a single hierarchical structure, but rather it is structured by an ECC analysis into a narrative stream.

FRUMP (DeJong 1979) produces representations of text by applying in a top-down fashion sketchy scripts (e.g. accidents and terrorist acts). It extracts from wire-service newspaper stories just enough facts to fill in the arguments of a sketchy script. Because the stories that FRUMP works with are so stereotyped, it could summarize text by using a set of fill-in-the-blank type summarization statements attached to each sketchy script. These techniques are by definition limited to stereotypical situations. Thick text includes many relationships between events which cannot be accounted for by a sketchy script, hence a sketchy script analysis can only account for some of the event relationships in a thick piece of text.

6. Summary and Conclusions

Thick text is composed of a number of embellished text descriptions that are enmeshed together. A piece of text is thick when it shows the reader, in detail, the events of the story, rather than telling the reader only of its skeletal structure. This paper has described a

summarizer called SSs that takes a thick piece of text and produces a skeletal structure for the events described in that text. SSs bases its summary of the story on an ECC analysis of the text. An ECC analysis of the text sorts through the events of the story, grouping together events based on their relative positions in an underlying conceptual network of events.

NEXUS and SSs have been applied to several examples of text, including a page and half folktale taken from a book of folktales ("The Clever Peasant and the Czar's General," Protter 1961), "The Xenon Story" (Wilensky 1980), and "The Czar's Three Daughters" (Lehnert & Loiselle, 1988). (See Alterman & Bookman for details). Experiments show NEXUS and SSs reducing the volume of text between 60% and 80%. The importance of this work, with regards to the problem of thick text, is that it reduces the gap between top-down theories of text concerned with issues of interest and salience (e.g. plot units or prototype points) and the complexity of the event relationships as they are depicted in the text.

References

Alterman, R. (1985). A Dictionary Based on Concept Coherence. *Artificial Intelligence*, Vol. 25, pp 153-186, North Holland.

Alterman, R. (1988). Event Concept Coherence. In D. Waltz (Ed.), Advances in Natural Language Understanding. Hillsdale, NJ: Lawrence Earlbaum (forthcoming).

Alterman, R. & Bookman, L. Some Computational Experiments in Summarization (in preparation).

Charniak, E. (1983). Passing Markers: A Theory of Contextual Influences in Language Comprehension. *Cognitive Science*, Vol. 7, pp 171-190.

Chun, H.W. AINET-2 User's Manual. Computer Science Department, Brandeis University, CS-86-126, Waltham MA, 1986.

Correira, A. (1980). Computing story trees. *American Journal of Computational Linguistics*, 6, pp 135-149.

DeJong, G. (1979). Prediction and Substantiation: A New Approach to Natural Language Processing. *Cognitive Science*, Vol 3, pp 251-273.

Lehnert, W. (1981). Plot Units and Narrative Summarization. *Cognitive Science*, Vol 5, no. 4, pp 293-331.

Lehnert, W. & Loiselle, C. (1988). An Introduction to Plot Units. In D. Waltz (Ed.), Advances in Natural Language Understanding. Hillsdale, NJ: Lawrence Earlbaum (forthcoming).

Norvig, P. (1987). Inference Processes and Knowledge Representation for Text Understanding. Phd Thesis, University of California at Berkeley, UCB/CSD 87/339.

Protter, E. (1961). A Children's Treasury of Folk and Fairy Tales. Channel Press.

Quillian, R. (1968). Semantic Memory. In M. Minsky (Ed.), Semantic Information Processing, MIT Press.

Rumelhart, D. E. (1975). Notes on A Schema for Stories. In D. G. Bobrow and A. Collins, (Eds.), Representation and Understanding, Academic Press.

Simmons, R. F. & Correira, A. (1980). Rule Forms for Verse, Sentences, and Story Trees. In N. Findler (Ed.), Associative networks: The Representation and Use of Knowledge in Computers. NY: Academic Press.

Van Dijk, T. A. (1976). Macro-structures and Cognition. In *Twelfth Annual Carnegie Symposium on Cognition*. Carnegie Foundation.

Wilensky, R. (1980). What's the point? In *Proceedings of the Third National Conference of the Canadian Society for the Computational Studies of Intelligence*.

Wilensky, R. (1982). Points: A theory of the structure of stories in memory. In W Lehnert & M. Ringle (Eds.), Strategies for natural language processing. Hillsdale, NJ: Lawrence Earlbaum.

Wilensky, R. (1983). Planning and understanding. Reading, MA: Addison-Wesley.

ON-LINE PROCESSING OF A PROCEDURAL TEXT

ANDRÉ RENAUD
CARL H. FREDERIKSEN

MCGILL UNIVERSITY

Abstract. The processing of sentences, propositions, and conceptual structures was studied using a task environment which required subjects to read, interpret on-line, and recall a procedural text while reading times were measured for each sentence. A declarative representation of the conceptual frame structure of the procedure expressed in the text, as well as propositional and syntactic analysis of sentences, provided variables that were used to predict these three sets of data. Results showed that properties of the procedural frame, as well as propositional density, and clause structure predicted reading times, recall, and on-line interpretation, and that reading times decreased when high-level conceptual frame processing increased. These results were interpreted as evidence for parallel on-line conceptual processing of sentences during input. As well, reading times for information near boundaries of conceptual structure reflected some buffering in comprehension.

The study of comprehension as an on-line process tries to identify the temporal locus of the various component processes involved in text comprehension and thereby to test hypotheses concerning the manner in which they interact. Component processes have been associated with different levels of representation of linguistic and semantic information, that is, syntactic structure, propositions and conceptual frame representations. These processing components are examined by using variables derived from theories of text representation to predict variables reflecting on-line and post-input processing.

One approach to this problem has emphasized the need for the elaboration of explicit models of representation to guide research. This research has focused on the cognitive representations that are manipulated, and the relation of these to prior knowledge stored in long-term memory. Data that are collected to test theories are usually based on recall or other post-input tasks. This approach has provided high-level descriptions of the processes that take place in comprehension but has had difficulty in identifying the temporal locus of processes during on-line or post-input processing.

A second approach has been more concerned with the description of component processes using real-time measures such as eye movements, word reading times and segment reading times for various units of text (Haberlandt & Graesser, 1985; Rayner & Carroll, 1984). The underlying assumption is that the time spent reading each of these units is an indication of the amount of processing each requires. Reading time measures are assumed to reflect processing of different kinds of text structure information predicted from a comprehension model. This is done by specifying comprehension components on the basis of a theory, and then in each text unit, identifying measurable variables corresponding to each component. The assessment of correlations between reading times and these variables are taken as indicators of the contribution of each component to the comprehension process. A major difficulty underlying these studies is in specifying how temporal measures relate to specific comprehension processes. Comprehension must be measured from variables that are theoretically motivated, and that are strongly tied to the component processes they are supposed to instantiate. Identifying theoretically-based variables requires a detailed model of comprehension. However, such a model has been lacking in

This research was supported by grants from the Social Science and Humanities Research Council of Canada and from the Natural Science and Engineering Research Council of Canada. Requests for reprints should be addressed to: André Renaud, Laboratory of Applied Cognitive Science, Department of Educational Psychology, McGill University, 3700 McTavish, Room B-199, Montreal, Quebec, Canada, H3A 1Y2.

most research applying this approach (Danks, 1986).

The present study investigated comprehension processes as they operate in real-time during the reading of a procedural text using measures of reading times and measures derived from the on-line interpretation of text units. In addition, post-input recall measures were also obtained. Models of propositional and procedural representation of the experimental text (Frederiksen, 1986), as well as clausal analysis of text sentences provided theoretically-based variables that were linked to the three following levels of processing of text units: a) syntactic analysis, b) proposition generation and inference, and c) the generation of a conceptual frame representation for the procedure. These models of the text information provided a way to link the two on-line and the post-input recall measures to the component processes associated with the processing of these kinds of information structure.

There is a considerable amount of information on the actual processing of the data structures corresponding to the first two components -- syntactic structure and propositions. As for the third, the conceptual frame representation of the knowledge presented in a text, an important development in this research has been the definition of precise models of conceptual processing in the form of semantic grammars similar to propositional grammar used to define and model the generation of propositions (Frederiksen, 1986; 1987).

Frederiksen (1986) has defined one such model as a procedural grammar that consists of a series of rules that can generate a network of nodes and links, called a "procedural frame", on the basis of a propositional analysis (Frederiksen, 1975). A node in a procedural frame is defined as a data structure that has the following information: a) an action that accomplishes a goal, produces a result, and can be executed by someone, b) a list of case information associated with that action, c) descriptions of related objects and states, d) tests that are conditions to the execution of this procedure, and e) various links among these nodes. Those links correspond to propositional structures in Frederiksen's model: the most common one in a procedural frame is the "part relation", in which a procedure is a component of another, creating a hierarchy. Other types of links, which may alter the strictly hierarchical aspect of the frame structure include algebraic, dependency, category, identity, and goal relations.

The use of a conceptual frame grammar as a tool for the analysis of procedural text also assumes that such a grammar is used by the reader. Reading is therefore considered as a rule-based process that generates and augments a model of the text. This initial representation is partial when only a few sentences have been read, but it guides and limits the reader's inferences. The representation is augmented and modified throughout the reading activity. The reader's knowledge of the possible procedural structure of the text helps in planning the reading and to limit inferences to those required to generate the model. We therefore hypothesize that the processing of the conceptual structure of the procedural text takes place during reading. Furthermore, we believe it is possible to assess the temporal locus of this comprehension component by measuring on-line interpretation and reading times for variables derived from this conceptual structure. Readers that are applying rules to generate the structure of a procedural text will actively search for procedural information, limiting propositional and syntactic processing. This will *reduce* reading times for sentences that contain propositions which contain information relevant to the procedural structure.

The first objective of this study was to determine if information about the conceptual representation of the procedure is processed immediately as the text units are read, or if a "buffering" strategy delayed this processing until "chunks" of propositional information have been generated. It is hypothesized that conceptual processes takes place on-line and that the resulting comprehension of the conceptual information is used to reduce the processing

load during reading. Consistent with the results of Haberlandt (1980) and Mandler & Goodman (1982), we also expect that some local buffering will take place.

A second objective was to test whether on-line interpretation of text sentences or multisentence "chunks" would reflect frame-generation processes (i.e., selective processing and inference) as has been found to occur during recall. It was hypothesized that the same frame variables would predict on-line interpretation as well as predict recall.

Third, it was hypothesized that an experimental manipulation of the time of interpretation would not affect these predictions.

METHOD

Text variables

The analysis of the experimental text on how to collect fossil bones, which was an unfamiliar procedure to subjects, yielded variables that correspond to the three main component processes. At the level of sentence processing our predictor was the number of major clauses and bound adjuncts (Winograd, 1983). Properties related to the generation of propositions were derived from the analysis of the experimental text using Frederiksen's propositional grammar (1975, 1986). This provided us as well with a measure of propositional density (number of propositions per clause per sentence) for the second component. The contribution of the third component process was studied using properties of the conceptual frame structure. Using Frederiksen's (1986) grammar for procedures, eight main sub-procedures were identified which explain how to localize fossils and remove them, how to protect, transport, and put them together again. This analysis then allowed us to classify text propositions according to various properties of this representation: 1) procedural versus non-procedural propositions, 2) the types of procedural propositions (actions, states, and links), 3) the types of non-procedural propositions (linked or not to an action), 4) the sub-procedure to which a proposition belongs, 5) propositions at the beginning, middle or end of a component, 6) and the level of a proposition in the hierarchy, which is measured by the number of links separating an action from the main procedure.

Subjects

Subjects were thirty four graduate students from the educational psychology department at McGill University. The procedure took approximately 30 minutes for each subject, tested individually.

Experimental conditions

The experimental conditions included two between-subjects factors: a) presentation condition of the text (no accumulation of sentences on the screen versus accumulation) and b) the execution of an on-line interpretation task (reading only versus reading and interpretation). When the interpretation task occurred with no accumulation, interpretation was forced after each third sentence; when it occurred with accumulation, the interpretation was under the subject's control.

Procedure

Subjects read the text while controlling the rate of presentation of the 29 sentences of the text which appeared one by one on the screen of a computer. Two kinds of data were available for all subjects: sentence reading times and propositions which were either recalled or served as the basis for an inference in the subject's protocol. For subjects in a task requiring on-line interpretation, measures of recall and inference were available for their on-line interpretation protocols as well.

A multivariate repeated measures analysis of variance was performed for each of the predictor variables, specified as the within-subjects factors. In each analysis, the experimental conditions were the between-subjects factors: presentation condition and task; and the within-subjects factor was one of the predictor variables. When the dependent variable was reading time, there was only one within-subjects factor; when the dependent variables were based on either the recall or on-line protocols, there was in addition a second within-subjects factor: response type (propositions recalled, recalled with local inference, and inference based on the

proposition). The effects of interest are main effects of each predictor variable, and any interactions of the between-subjects variables with the predictor variables.

RESULTS AND DISCUSSION

Effects of experimental conditions

One objective of this study was to establish that the use of an on-line interpretation procedure did not affect the normal comprehension of the text. There were no significant main effects of these factors in any of the thirty analyses of variance. Only four borderline significant interactions of experimental task were found, which is less than would be expected due to chance. Since there was no effect due to the "chunk versus forced" interpretation task condition, there is no evidence that chunking occurs as a general buffering strategy.

Consistency of recall and interpretation data

As may be seen in Tables 1 and 2, significant effects occurred for all but one of the predictor variables for the recall data. These involved (with the exception of "sub-procedure boundaries") both significant main effects and interactions with response types. Thus, there was selective processing associated with these variables as well as differential inference. When the on-line interpretation data were predicted, significant main effects or interactions occurred for all but the "main sub-procedures" variable. In the case of "non-procedural propositions" and "level in the structure", the effects were found entirely in interactions involving response types (i.e., extent of inferences). Thus the effects show up more in the inferences than in recall of propositions. These differences between selective processing are to be expected since in the on-line tasks, the propositional information was more recently available to the subjects and therefore easier to recall. However, other than these differences, the results were identical with recall and on-line interpretation data. Thus the results were clearly consistent with the "on-line" model, that is, selective processing does occur on-line, and they indicate that the use of the on-line interpretation task did not influence the normal processing of the text.

TABLE 1
Summary of Significant Effects for Predictor Variables.

	Recall		*On-line*		*Reading time*
	Main	*Inter*	*Main*	*Inter*	*Main*
Conceptual structure					
Types of Propositions	0.0001	0.0001	0.0001	0.0001	0.05
Procedural Propositions	0.0001	0.0001	0.0001	0.0001	0.0001
Descriptive Propositions	0.0001	0.001	n.s.	0.05	0.0001
Main Sub-Procedures	0.0001	0.0001	n.s.	n.s.	0.0001
Sub-Procedure Boundaries	n.s.	0.05	n.s.	0.001	0.0001
Level in the Procedure	0.01	0.0001	n.s.	0.01	0.0001
Proposition level					
Propositional Density	0.0001	0.01	0.01	0.01	0.0001
Sentence level					
Number of Clauses	0.0001	0.05	0.001	0.01	n.s.

Effects involving sentence and propositional levels of processing

The predictor variable reflecting proposition-level processing produced significant effects on all dependant measures, while the variable reflecting sentence processing had no significant effects for reading times. Propositional density had a significant curvilinear relationship to reading time (figure 1), in which a plot of reading time against density was flat until density levels over seven;

after a density of seven propositions per clause, the curve positively accelerated[1].

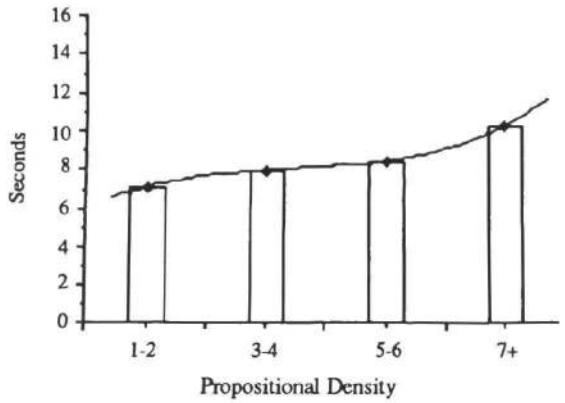


Figure 1. Reading Times for Sentences having Different Levels of Propositional Density

Effects involving processing of conceptual frame information

Procedural information

Reading times for sentences encoding procedural propositions were faster than for sentences encoding non-procedural propositions. This reduction in reading times indicates that when information from the conceptual structure is encountered there is a facilitation effect. We also observe that sentences with propositions representing actions, which are the most important in a procedure, are read faster than sentences with states or links. These results are consistent with the hypothesis that a procedural model is generated on-line during reading rather than according to a general "buffering" strategy in which chunks of propositions are stored in a buffer prior to frame-generation.

Non-Procedural Descriptive Propositions

We can find in the text, along with the procedural information, descriptions that are not directly pertinent for the execution of the general procedure. These descriptions are formed by a set of propositions that can be linked to a procedure (e.g., a description of the location where fossils can be found), or present information that is not directly pertinent (e.g., what a fossil is made of). Longer reading times for sentences with propositions that are non-linked were found to be an indication of the greater difficulty of integrating this information into the procedural model of the text.

Procedural components

Sentences containing propositions from the eight sub-procedures varied significantly in the average time it takes to read them. These differ in many ways: a) the number of propositions they contain, b) the types of links among procedures, c) the number of levels spanned in the hierarchy, and d) the manner in which the propositions that compose them are linearized in the text. This last property, the linearization of a conceptual structure, is a fairly complex aspect of this level of processing since the various propositions of a sub-component can be combined in the text in various ways. In fact, many texts could theoretically be generated from the same conceptual structure. While the differential processing of procedural components is supported by the pattern of reading times, it is impossible to order components in terms of their complexity.

Procedural component boundaries

Interaction effects with response types were found for both recall and interpretation data, but there was no main effect. As readers progressed towards the end of a component, they were making more inferences and less recall of propositions. The significant increase in reading time for sentences that contain propositions at the end of components (figure 2), as well as the increase of inferences for those same propositions (figure 3), reveals that integrative processes occur at the end of a component. This phenomenon also has been documented in research on episode units in narration of Haberlandt (1980), Haberlandt, Berian, & Sandson, (1980), and of Mandler & Goodman (1982).

[1] One of the three sentences that should have been included in the propositional density category of 7 or more propositions per clause had to be eliminated. Since it consisted of a list of tools, subjects often tried to memorize the list by rereading, thus using a strategy that increased reading times for reasons other than the processing of both propositional and syntactic aspects of the sentence.

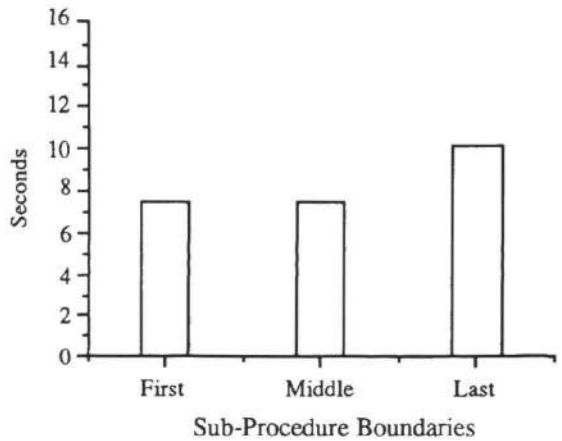


Figure 2. Reading Times for Sentences with Propositions at the Sub-Procedure Boundaries

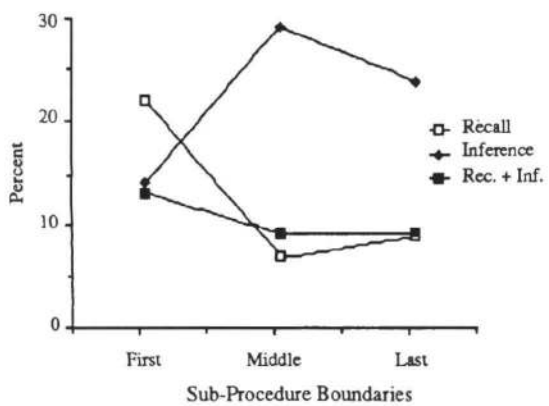


Figure 3. On-Line Responses for Propositions at the Sub-Procedure Boundaries

Levels in the procedural hierarchy

The number of links separating the main procedure from its sub-procedures indicated the presence of four levels in the procedural hierarchy. A comparison of the averages of sentence reading times for this variable revealed a significant effect. Sentences containing propositions at the first and last levels were read faster than those at the second or third levels, which are read at a slower rate. A trend analysis of this curve revealed a strong cubic component, which is presented in figure 4.

This phenomenon can be interpreted on the basis of characteristics of the conceptual representation of the text: when a procedural node must be linked to the rest of the conceptual structure, reading time should increase as a function of the number of links that must be generated. Thus, at the first level we find the propositions that represent

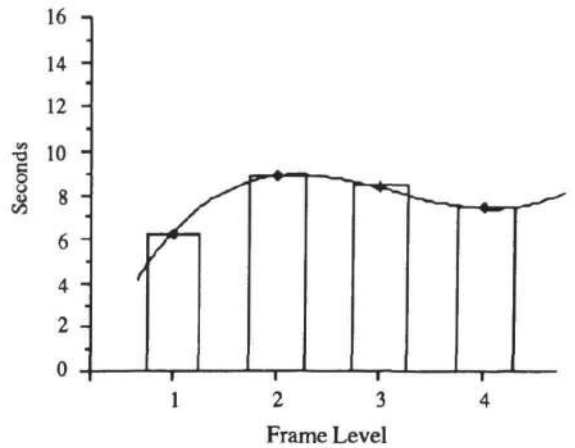


Figure 4. Reading Times for Sentences having Propositions at Different Levels in the Procedure

the eight main sub-procedures, each having a single link to the main procedure. Sentences containing propositions at this level should be read faster than those at the second level. There should be no difference in time required to read sentences with procedures at the second and third levels since they have to be linked to approximately the same number of nodes: with the superordinate node and with one or many subordinate node(s). Reading times should decrease for sentences with propositions at the last level in the hierarchy since a single link exists between each procedural node and its superordinate node.

These results seem to contradict those of Cirilo & Foss (1980) who found that reading times increased as a function of propositions in the levels of the hierarchy. This difference is due mainly to the fact that their hierarchy is based on the overlap of arguments in memory when processing propositions (Kintsch & van Dijk, 1978), which reflects the coherence of the text rather than its conceptual representation (Cirilo & Foss, 1980).

CONCLUSION

All variables specified using the conceptual structure of the procedural text

produced significant effects for reading times as well as on recall and (with the exception of frame components) on the on-line interpretation. These results confirm the hypothesis that selective conceptual processing of text information occurs on-line. However, conclusions concerning the locus of conceptual processing of a text require analysis of the relationship of processing time to propositional information actually generated on-line and during recall.

Analysis of the recall and on-line data confirmed that a decrease in reading time for sentences that contain procedural information was accompanied by an increase in high-level inferences. These results replicate those already obtained for measures of recall and inference with the same text, but with younger children (Frederiksen, 1987).

This covariation of reading time with propositional and conceptual information generated is consistent with a parallel and modular model in which lower-level processing is reduced when information is processed conceptually.

We observed, however, a different pattern for sentences with propositions at the end of a sub-procedure. In this instance an increase in reading time correlated with an increase in high-level inference. We can interpret this as an indication that sub-procedures are processed on-line, but that they are not integrated immediately. Rather, it seems to reflect a buffering strategy in which propositional information is stored temporally until the end of a component is reached and then the information is integrated into the procedural frame (Kintsch & van Dijk, 1978).

The effect of the level of a proposition in the hierarchy on sentence reading time also suggests that the process of generating complex conceptual networks is a distinct process that is reflecting the nature of the network.

Finally, these results involving the effect of conceptual structures on the on-line processing and recall of sentences are entirely compatible with the rule-based model, that is, the notion that a reader generates a conceptual model of a text by applying semantic rules to the text information. In addition, reading time was found to reflect the network properties of the structure being generated as well as the nature of the rules being applied.

REFERENCES

Cirilo, R. K. & Foss, D. J. (1980). Text structure and reading time for sentences. Journal of Verbal Learning and Verbal Behavior, 19. 96-109.

Danks, J. H. (1986). Identifying component processes in text comprehension: Comment on Haberlandt and Graesser. Journal of Experimental Psychology: General, 113, 193-197.

Frederiksen, C. H. (1975). Representing logical and semantic structure of knowledge acquired from discourse. Cognitive Psychology, 7, 371-458.

Frederiksen, C. H. (1986). Cognitive Models and Discourse Analysis. In C. R. Cooper & S. Greenbaum (Eds.), Written Communication Annual: An International survey of research and Theory, Vol. 1: Studying Writing: Linguistic Approaches (pp. 227-267). Beverly Hills, CA: Sage.

Frederiksen, C. H. (1987). Text Comprehension in Functional Task Domains. In D. Bloom (Ed.), Functional Literacy Activity in Education: Cognitive and Social Approaches. Norwood, N. J.: Ablex.

Haberlandt, K. (1980). Story grammars and reading time of story constituents. Poetics, 9, 99-118.

Haberlandt, K. Berian, C. & Sandson, J. (1980). The episode schema in story processing. Journal of Verbal Learning and Verbal Behavior, 19, 635-650.

Haberlandt, K. F. & Graesser, A. C. (1985). Component processes in text comprehension and some of their interactions. Journal of Experimental Psychology: General, 114, 357-374.

Kintsch, W. & van Dijk, T. A. (1978). Toward a model of text comprehension and production. Psychological Review, 85 (5), 363-394.

Mandler, J. M. & Goodman, M. S. (1982). On the psychological validity of story structure. Journal of Verbal Learning and Verbal Behavior, 21, 507-523.

Rayner, K. & Carroll, P. J. (1984). Eye movements and reading comprehension. In D. E. Kieras & M. A. Just (Eds.) New Methods in Reading Comprehension Research. Hillsdale, N. J.: Lawrence Erlbaum Associates.

Winograd, T. (1983) Language as a Cognitive Process: Syntax. Reading, M.A.: Addison-Wesley.

Understanding Stories in their Social Context

Eric Domeshek

Yale University
Department of Computer Science
New Haven, CT 06520-2158

Abstract

Stories concerning multiple agents with interacting goals and plans are difficult to understand; the task can be simplified however, if a program is given sufficient knowledge of social structures. Representations of social aspects of the story may also be necessary components of a satisfactory understanding of the story; here, we consider the distinctions which must be representable to support the task of *advice-giving* in the social-domain.

This paper elaborates on established goal taxonomies in order to capture distinctions among goals embedded in a social context; these distinctions serve both as the basis for choosing advice and as inferential shortcuts. It goes on to explore ways in which such social goals can be predicted from a detailed understanding of the conventions defining social structures linking the various agents. *Social-units, social-situations*, and *triangles* are introduced as conceptual structures which organize interpersonal-themes. The goals predicted by these themes provide a focus for motivational and impact inferences which would otherwise be lacking. These and related structures also allow some direct predictions of actions, as when a social-situation provides scriptal specification of action sequences or when a *contract* underlying a social-unit licenses specific *recourses* in response to obligation failures.

I. ABBY the Advice Giver

Story understanding has progressed from understanding scripty stories [2], [3], to simple plan based sequences [14], to more complex stories of multiple agents with interacting plans and goals. Relatively little attention has been paid to representing the details of the social context in which the agents act out the story; typically, programs either assume a fixed cast of characters with fixed goal structures [1], or limit the range of possible relationships among agents [4]. The first approach is an unrealistic simplification in most cases; the second throws away a potentially rich source of expectations.

Embedding agents in a social context should actually simplify the task of understanding by providing expectations about likely goals and plans; so long as such expectations are not consistently wrong, they will serve to effectively restrict the amount of inference required to make sense of the story. Furthermore, explicit representation of the story's social context may actually form a necessary part of a satisfactory understanding for some purpose [11].

This paper reports work done in the context of the ABBY project. ABBY is a story-telling advice-giver for the social domain. Named in honor of the legendary Abigail VanBeuren of the syndicated newspaper column *Dear Abby*, ABBY reads[1] stories describing lovelorn situations and responds with advice couched in the form of a story. Essentially, ABBY is half of a case-based planning system [9], [5], [6] — modeling memory, problem understanding, and case retrieval. Its ability to find relevant cases in memory depends on a rich representation both of the cases in memory and of the input case.

This research was supported in part by the Advanced Research Projects Agency of the Department of Defense and monitored by the Office of Naval Research under contract N00014-85-K-0108.

[1] Lacking a parser, the current implementation of ABBY does not actually *read* anything, rather it is given a representation of the input in terms of such abstract categories as could easily be extracted from a natural language version of the story. That is to say *marriage of a man and a woman* as opposed to *rushed marriage of frivolous high-school sweethearts*.

The program operates by viewing any new input as being like some other situation it has already encountered; it sorts inputs into the categories defined by its experiences. When ABBY tells a story as advice, it is not simply because a story is a good way to communicate a point (which it is), but because that past experience contributed significantly to its understanding of the input.

Consider this example of a story handled by ABBY:

> Dear ABBY,
>
> My boyfriend, Joe, and I have been going together for 3 years, and we are just about to graduate from high-school. We love each other more than anything, and we want to get married. My parents, though, say we should wait until Joe gets some kind of job and saves up enough money. We think that since we love each other we should be together. Jobs and money will work themselves out. How can we convince our parents?
>
> - Already Ready

Appropriate advice for this input could come from a reminding of another case of rushed marriage of high-school sweethearts.

> Dear Already Ready,
>
> I knew a couple, Brenda and Eddie, who rushed into marriage as soon as they got out of high school.[2] They had been going together for years and were the most popular couple in their class. Even so, everyone told them they would never make it in marriage. All they knew about was the fun they had together. Neither one had ever had to work for any more than pocket money; earning enough to buy gas and malteds isn't the same as supporting a household. They learned that fast enough. Their marriage lasted only a few months — just about as long as their savings.
>
> - ABBY

ABBY must be judged on the quality of its remindings. Within the paradigm of case-based reasoning, expertise results from massive experience; a memory stocked with problems carrying observed or generated solutions provides a reasonable guess at a good plan in any new situation. Though the current implementation has but on the order of a hundred cases in memory, and is only planned to contain on the order of a thousand cases, the intent is to model a human expert's memory with its tens or hundreds of thousands of cases. If we imagine a system with a memory as rich in experience as the real "Dear Abby", or even as the average adult, we can begin to ask what features would differentiate one experience from another. What makes one episode a better reminding than another? This raises the question of representation. If one reminding is really better advice than another, then the differences that make the difference must be representable.

ABBY's representational ontology includes many of the conceptual classes introduced by earlier work in story understanding [10], [1],[8],[4],[9]. There are individuals representing *states, actions, things, agents, themes, goals, preferences, affects*, plus concepts for *temporal* and *causal* linkages, as well as larger aggregate structures such as *MOPs, TOPs*, and *XPs*. In keeping with ABBY's focus on classification, each of these broad classes dominates a detailed hierarchy of specialized variants.

These representations must facilitate detailed analysis of the story's facts in terms of their impacts on characters' goals, with an emphasis on instances of goal failure; failed and threatened goals are the problems about which ABBY gives advice. The major difficulty for ABBY is that the space of possible goals is nearly unlimited, and the linkage from reported facts to those goals may be arbitrarily indirect; with enough ingenuity, nearly any fact can be shown to impact nearly any goal. The search for relevant goals — those goals that either motivated story actions, fortuitously benefited from those actions, or suffered because of them — must be limited.

Limiting this search requires predictive knowledge of what types of goals the characters are likely to actually hold, and what types of goals are most likely to be affected by reported and infered facts. This paper focuses on some of the representational machinery developed to supply these expectations. Following [10], ABBY uses several classes of *themes* as explanatory and predictive organizers of goals. After elaborating a taxonomy of goals specifically

[2] This story is actually adapted from a song by Billy Joel, *Scenes from an Italian Restaurant* off his album *The Stranger*. Though I don't think I've listened to it in years, in the context of this work, I've lately had trouble keeping it out of my head; I mention this to credit Billy Joel, and as evidence of how memorable a good story can be.

for social contexts, this paper discusses how ABBY derives agents' themes from membership in *social-units* and ad-hoc groups, and from participation in *social-situations* and *triangles*.

II. Goals: What do People Want Anyway?

Much work has been done in elaborating goal taxonomies. Our point of departure is the taxonomy proposed in [10]. Among the distinctions made there, the most important for present purposes are those between *satisfaction, achievement*, and *preservation* goals.

Satisfaction goals are those that require repeated attention; they may be satiated at one time, but eventually — as originally proposed, on some biologically determined schedule — they will again demand attention. In ABBY, almost all goals are roughly of this form, subject to the generalization that we are no longer just talking about biological drives as for food, sleep, or sex. Under this broadened interpretation, almost all possession goals to fit here. For many things it is not enough to just possess them once; money for instance gets spent, and then you need to get more. This is contrary to the original classification in [10] which counts possession goals among the achievement goals. The prototypical achievement goal though is something like *wanting to marry somone*; that is, its object is a state that you achieve once and then maintain. This is the form in which achievement goals appear in ABBY.

Finally, preservation goals as presented in [10] are reactive goals — they come into being when an agent percieves a threat to some desired state. In ABBY, this notion is generalized by introducing *maintenance* goals. Maintenance goals always exist for a desired state; they account both for the spawning of P-goals in response to a threat, and for anticipatory plans aimed at maintaining the desired state. Consider the goal *maintain-health*; the standard plan of scheduling regular checkups with doctors and dentists is intended to maintain health without needing to experience an imminent threat.

A. Social Goals

In a social context, important goals include what one wants for another, what one can do for another, what one can get from another, and what a group can do together for some members, all members, or non-members. If we start with some basic biological goal state such as *satiate-hunger — X want's X's hunger satiated* — social variants of this goal would include:

- X wants Y's hunger satiated
- X wants Y's hunger satiated by X
- X wants Y's hunger satiated by Y
- X wants Y's hunger satiated by Z
- X wants X's hunger satiated by Y

Such distinctions make sense when we consider beliefs such as: *parents ought to feed their children.* Now a child can go begging in the street and get something to eat — that will take care of the child getting fed — but it does not meet the original goal specification; we still hold that the parent has failed to meet an expected obligation. Considered from the child's perspective, this may simultaneously be a goal success and a goal failure: it didn't just want to eat, it wanted to be fed, and it wanted to be fed by its parents. These distinctions matter to the extent that failures of goals and obligations matter. In the social world they matter quite a bit; we would understand if the child held a grudge against its parents for this percieved failure, and if this had some future consequences.

Not surprisingly, the same social variants appear when dealing with overtly social goals, goals less directly coupled to biology. Consider the goal to *start dating.* It is important to notice that wanting to get a date is different from wanting to date someone in particular. An advice-giver must notice the difference between X wanting to start dating Y and X's mother wanting X to start dating Y. It is reasonable to consider how X feels about other agents' goals for him; a mother with such goals might be a particular type of mother that we know something about — perhaps we have specific advice for how to deal with pushy prying mothers mixing into their childrens' social lives.

Social goal variants can be further differentiated based on the relationships holding between the X's, Y's, and Z's. We have specific knowledge about the conditions under which certain goals are required, appropriate, expected or forbidden; those conditions often depend on the relationships holding between agents. Consider the mutual goal *having sex with someone.* If the agents involved are husband and wife this is a normal goal, in fact it is an obligation of marriage. If the pair are lovers it

may not be obligatory, but its certainly normal. If the agents are brother and sister — or in fact any close relation — the desire is considered taboo.

There are still other important distinctions among goals in social contexts. When X has a goal with respect to Y, we frequently can predict whether Y wants X to pursue that goal. Parents want their children to be fed; children want their parents to act on that goal. Such goals are *desired* goals; if X satisfies the goal with respect to Y, Y is also having a goal satisfied, if X fails, Y has a failure as well. Desired goals are a shorthand that facilitate noticing impacts, both positive and negative; knowing that a theme-partner was helped or hurt also aids in propogating affect inferences.

Other goals may be classed as *balanced* goals. These are paired goals where X has a goal with respect to Y, and Y has a similar goal with respect to X. As an example consider the goal *ensure-happiness* that normally holds between romantic partners. When ABBY notices X's goal is relevant, it is worth considering if Y's balancing goal is also relevant. For instance, when the goals are both balanced and desired, failure of one may, under the strategy of "tit for tat" lead to failure of the other; if X fails to make Y happy, Y may refrain from making X happy.

Closely related to balanced-desired goals are what Seifert has called *mutual goals* [11], that is, goals whose object states, by definition or by social convention, require the cooperative efforts of multiple agents. Examples include *spending time, conversing*, and *starting or maintaining a relationship*. Seifert has elaborated in great detail inferences and strategies specific to mutual goals [12].

B. Subsumption Goals

A large class of social goals can be viewed as goals for abstract possession. This is intended as a generalization of normal possession goals, so that we can speak not only of an agent wanting tangible items like food, clothing, or money, but also of intangibles like company, respect, and love. This notion of abstract possession — of *having* something — suggests the companion social goals of *giving* and *getting* these objects. Again, there are many possible social variants:

- X wants X to have O
- X wants Y to have O
- X wants X to get O
- X wants Y to get O
- X wants X to give O
- X wants Y to give O
- X wants X to give O to Y
- X wants Y to give O to X
- X wants Y to give O to Z

Possession of an object O is generally desired if O is instrumental to some other goal. Having food is a *goal subsumption state* that helps in achieving recurring goal to *satiate-hunger*. A state subsumes another goal-state if achieving the subsumption state makes it trivial to repeatedly achieve the goal-state [13]. In such a situation, it makes sense to pursue the subsumption-state as a goal in its own right. Thus *haveing-food* becomes a common goal even though milk in the refrigerator is doing no one any immediate good.

The number of possible tangible and abstract entities worth possessing is huge. This section only sketches the outlines of a potentially vast class of common goals; they form the basis for the claim that most of the goals ABBY is concerned with behave like satisfaction goals. Just as it's hard to make it through life on one week's pay, so too is it tough to get by on one unrepeated display of affection. Agents are constantly concerned with ensuring these possession goals get satisfied.

Just as achieving possession of some object can be a subsumption goal, so ensuring a reliable stream of that object — so that you always possess enough of it — can also become a subsumption goal [13]. This is the basis for ABBY's notion of *social-unit*: a long-term contractual arrangements with other agents designed to provide needs on a continuing basis. Joining a company as an employee is the most obvious example of entry into a social-unit serving to subsume possession goals. Less tangible possessions accrue from becoming someone's friend, but companionship, support, and advice are valuable nonetheless. Of course there are social-units that are not usefully explained this way; for example it does not account for why the average conscript joins the army.

Social-units then can be viewed as plans for subsuming common goals. Adopting those plans engenders other goals; here we hark back to the earlier analysis of achievement goals. People spend much of their time pursuing the *initiation, maintenance,*

termination, and *restoration* of social-units, or trying to *join* and *quit* other larger social-units. All these (except maintenance goals as described earlier) are achievement goals. In lovelorn situations, it is often these high-level achievement goals that are central to a story.

III. Social-Units and Contracts

We organize the social world by placing people into social-units. They are so pervasive and powerfully predictive that they or the roles and relationships that come with them are almost always explicitly mentioned when discussing people. Who is John? He is Mary's husband, Jimmy's father, Ethel and Eddie's son, an employee of International Widget, a devout church-goer, a good neighbor. That sort of capsule summary can immediately relate John to anyone of importance in his world.

It is interesting to note that for a wide range of social-units we have specific words or phrases to denote the units, the roles, and even the relationships between role-fillers. As suggested, these are very important concepts frequently worth communicating — they define the most important features of any individuals social landscape in powerful shorthand.

Social-units range in size from diads to cultures. On the smallest scale, we live in a world of *acquaintances, friendships*, and *romances*. In the midground we define ourselves with respect to our *families, neighborhoods*, and *workplaces*. In the larger world, we are members of *professions, religions*, and *nations*.

Each member plays some *role* in a social-unit; members relate to each other differently depending on their respective roles. Within a family there are roles for *parents (mother, father)*, and *children (son, daughter)*. The fillers of these roles are related to one another as *parent-of (mother-of, father-of)* or *child-of (son-of, daughter-of)*.

We know quite a bit about what it means to be someone's parent. Such relationships function as *interpersonal-themes*; that is, they lead to specific goals of one agent involving the other. Every social-unit defines a set of interpersonal-themes. ABBY can use these to predict what social goals are likely to be active between agents. Normal parents want to provide material, social and emotional support for their children. Children want those things provided to them by their parents.

As suggested before, some goals have a special force in that they are obligations. Even if there are parents who in some sense don't *want* to feed their children, we still expect them to act as if they hold the goal. Claiming that an obligation leads you to want what you don't want is not really a contradiction; goals, as representations, are primarily predictive and explanatory constructs. If we believe an agent has an obligation it is reasonable to try to see actions as attempts to meet the obligation; if we see someone acting in a way that satisfies an obligation it is reasonable to use the obligation as part of the explanation for the action. That is all that a goal means. We do however need to be able to notice when an obligation, though fulfilled is resented — when the goal state is not really wanted.

Reciprocal relationships inherent in social-units, such as parent and child, generally imply *contracts* between the parties; we might take a long-term contractual basis as one of the defining features of social-units. Contracts are what confer special status on obligations. As representational structures, contracts package the reciprocal interpersonal-themes and provide predictions when obligations are not met; contracts license *recourses* — legitimate forms of retribution for obligation failure. This accounts for why obligations may be effective goals even though the goal-state is not in itself desirable; in some cases it is only fear of such recourses that leads a sposue to visit with the in-laws.

Trappings of romance and true love notwithstanding, marriage is a case where a contractual analysis of the relationship is openly adopted in many societies. For example husband and wife are supposed to sleep with each other exclusively; should one violate that part of the contract, the other has recourse to the courts where they can sue for divorce and may be awarded alimony. Violating just about any aspect of the marriage contract leads to the same recourse; in some societies, failure to produce male sons is likewise legitimate grounds for divorce. Dissolution of the underlying social-unit that invoked the contract is the most blunt and sweeping, but also one of the most common recourses. Still there are others; spouses may seek counciling before resorting to divorce.

The prototypical contract is an agreement entered into freely by equals, in which each party

agrees to assume certain obligations such that both parties benefit in some way. Many social-units deviate from this ideal in some respects, yet are still usefully thought of as contracts. This is particularly true when the contract itself is put into a social context; the contract may be enforcable by — that is recourses may be administered through the agency of — a higher authority. We see traces of such outside influences even in relatively prototypical social-contracts like that implied by marriage.

The social contract between parent and child is a good example of one which deviates significantly from the contract prototype. In the first place, the child never asked to enter into that contract. Furthermore, the parent has absolute power over the child. This is similar to the situation of the conscript alluded to earlier. In both cases, enforcement of obligations of the stronger to the weaker party depends on an implicit contract with some higher authority — that of the state. Unfortunately for the conscript, the army and the state are usually the same thing (or worse, the army may be superior to the state).

IV. Scripts as Social-Situations

The notion of social-unit bears an interesting relationship to the older notion of *script* [10]. Many scripts are of a form that might reasonably be called *social-situations*; that is they are spatio-temporally bounded, socially prescribed sequences played out by agents, temporarily related by their entrance into a short-term contract. Along with scripts came the notion of *role-themes* — organizers for the goals agents adopt when they enter into their script roles. Viewed as social-situations, scripts should spawn interpersonal-themes, and in fact it turns out that in social scripts, the goals organized by role-themes are in fact social goals: one agent wants something from another, or wants to do something for another.

In a restaurant, the customer has a temporary contractual relationship to the server; they have goals and obligations with respect to one another, and there are sanctioned recourses should either violate parts of that contract. Beyond the specific actions and their specific sequencing, we know that the server is supposed to provide service in a timely and curteous manner, the customer is supposed to behave civily and pay his bill. If the service is unsatisfactory, the customer may refuse to leave a tip. If a customer is offensive or abusive, the server may call the manager to have the customer removed. Again, the parallel to the case of a social-unit with an unbalanced power relationship is striking; a younger child has the same sort of recourse to its parents' authority when frustrated by an older sibling's misbehavior.

What counts as a short-term social-situation and what as long-term social-unit will usually be clear enough — long-term relationships generally lack a pre-specified termination conditions (short of death). Business units might be an exception: employment is generally long-term, but often has well-specified termination conditions. The scriptiness of a social-situation inheres in the stable predictability of its settings, sequence, and specific actions.

Just as several social-units can link the same agents, so several social-situations can be simultaneously active, as can a mixture of social-units and social-situations. Any of these combinations can generate interacting goals. A classic social-unit interaction stems from taking a son into the family business, which forces father and son to become boss and employee. Going to a restaurant where one of your friends works thrusts one among equals into servant status — a social-situation impinges on a social-unit.

V. Efficiency and Adequacy

This paper has presented a variety of representational constructs for use in understanding stories from the social domain. It has argued for particular distinctions and particular classes of concepts on two grounds: efficiency and adequacy. Efficiency implies that having a particular representation makes expectations available which effectively focus subsequent processing. Adequacy implies that without noticing a certain distinction, the program cannot succeed at its task.

In the task of story understanding, the major source of intractability is the need to notice the import of facts. What motives lay behind any given fact? Which agents are affected by that fact and how are they affected? Answering these questions normally requires hypothesizing goals for the agents and using causal knowledge to determine the impact of the fact on those goals. Detailed knowedge of the social domain can simplify the task of understanding complex stories in which multiple agents hold interacting goals. Recognizing the social con-

text which relates characters, provides exactly the kind of expectations that are most needed: expectations about which goals are active and relevant. In addition, there are some situations in which social knowledge can provide even more specific predictions about likely action sequences.

Many of the distinctions introduced in the classification of goals serve not to speed processing, but to make it effective. To succeed in the task of advice giving, a system must be able to detect and represent some very subtle properties of situations. This paper pointed out some of the social variants of goals to which an advice giver must be sensitive. Noticing specific types of goals may hint at particular types of relationships. Specializing the system's understanding of the relationship between agents may tighten the predictions about what types of problems are likely to arise, or allow better predictions about what types of solutions will work — that is, it may lead to better advice.

This paper did not discuss the system's understanding of an agent's character, but specialized stereotype information can yield the same sorts of advantages. The same holds true with specialized types of goals, and interpretations of actions. Clusters of such specialized concepts hang and work together. A family may be like the Waltons: the mother may be a saint, the father stern but fair, their relationships to the children close and loving; then again, a family may be more like some nightmare version of the Bunkers: the mother a dizzy simpleton, the father a bigoted tyrant, and their relationships to the children based on manipulation and domination. People know about all these types and know which ones tend to go together; pinning down one, helps to pin down the other. Getting the whole picture suggests specific advice suited to the situation. The representations presented here form part of that sort of complete picture, and as such are a part of what ABBY must understand to give good advice.

References

[1] J.G. Carbonell. *Subjective Understanding: Computer Models of Belief Systems.* PhD thesis, Yale University, 1979.

[2] R. Cullingford. *Script Application: Computer Understanding of Newspaper Stories.* PhD thesis, Yale University, 1978.

[3] G.F. DeJong. *Skimming Newspaper Stories by Computer.* PhD thesis, Yale University, 1979.

[4] G.D. Dyer. *In-depth Understanding: A Computer Model of Integrated Processing For Narrative Comprehension.* PhD thesis, Yale University, 1982.

[5] K.J. Hammond. *Case-based Planning: An Integrated Theory of Planning, Learning and Memory.* PhD thesis, Yale University, 1986.

[6] J. Kolodner. Extending problem solver capabilities through case-based inference. In *Proceedings of the Fourth International Workshop on Machine Learning*, pages 167–178, University of California, Irvine, Morgan Kaufman Publishers, Inc., Los Altos, CA, June 1987.

[7] N.I. Adams Rees, J.A. and J.R. Meehan. *The T Manual.* Computer Science Department, Yale University, 1984.

[8] R.C. Schank. *Dynamic Memory.* Earlbaum, Hillsdale, N.J., 1982.

[9] R.C. Schank. *Explanation Patterns: Understanding Mechanically and Creatively.* Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[10] R.C. Schank and R. Abelson. *Scripts, Plans, Goals, and Understanding.* Earlbaum, Hillsdale, N.J., 1977.

[11] C.M. Seifert. *Mental Representations of Social Knowledge: A Computational Approach to Reasoning About Relationships.* PhD thesis, Yale University, 1987.

[12] C.M. Seifert. Planning principles specific to mutual goals. In *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, COGSCI-87, Seattle, August 1987.

[13] R. Wilensky. *Planing and Understanding.* Addison-Wesley, Reading, MA, 1983.

[14] R. Wilensky. *Understanding Goal-Based Stories.* PhD thesis, Yale University, 1979.

Context effects in the comprehension of idioms

PATRIZIA TABOSSI CRISTINA CACCIARI

DIPARTIMENTO DI PSICOLOGIA, UNIVERSITA' DI BOLOGNA, ITALIA

An idiomatic expression is generally conceived of as an expression whose meaning is not deducible from the meanings of its constituents (Katz, 1973). Thus the challenging question for the standard view of literal language comprehension is how do people comprehend idioms. That is, how and when do people derive the idiomatic interpretation of a sentence on the basis of the literal interpretation of the words composing it.

So far, two main hypotheses have been proposed by psycholinguists. Swinney and Cutler (1979) had subjects judge the meaningfulness of idiomatic strings such as break the ice as well as literal strings obtained by changing a part of the idiom, e.g., break the cup. Idiomatic strings were judged faster than their literal counterparts. Swinney & Cutler interpreted this result as showing that idioms are stored in the mental lexicon as individual global items, that their retrieval is initiated as soon as the first word of an idiomatic string is encountered and this occurs in parallel with the computation of the literal meaning.

Gibbs (1980, 1986) proposed an alternative hypothesis, the Direct Access Hypothesis, even though the pattern of his results is consistent with Swinney and Cutler's Lexical Representation Hypothesis. Gibbs presented subjects with short stories ending with an idiomatic expression (e.g., he kept it under his hat). The subjects judged the meaningfulness of a sentence, following the story, paraphrasing either the idiomatic meaning or the literal meaning or an unrelated sentence. The story biased subjects either toward the idiomatic meaning of the sentence or the literal one. Gibbs found that subjects were faster at judging the idiomatic paraphrase than the literal and the unrelated regardless of the type of story. He interpreted the results as demonstrating that people automatically tend to analyze the conventional, idiomatic meaning of idioms and that the literal meaning of the sentence need not be computed at all.

Despite those two studies, evidence supporting one or the other hypothesis is far from clear, primarily because the Lexical Representation Hypothesis and the Direct Access Hypothesis generally lead to the same predictions (Ortony, Schallert, Reynolds & Antos, 1978; Estill & Kemper, 1982; Glass, 1983). There seems to be general agreement that the figurative interpretation of an idiom is formed faster than its literal interpretation, both in and out of context. However, most of the studies cited above employed meta-linguistic judgments of

comprehension processes, and so it is unlikely that such indirect measures could tap the fast processes that underlie language understanding.

More recently, Cacciari and Tabossi (1987) proposed the Configuration Hypothesis. In three experiments using an on-line paradigm (cross-modal lexical priming), they had subjects listen to neutral sentences containing idiomatic expressions. These expressions were idioms that did not have any clear literal counterparts (e.g., shoot the breeze). In experiments 1 and 2, a visually presented target word appeared immediately at the offset of the idiomatic expression, e.g., at heaven in the sentence After the excellent performance the tennis player was in seventh heaven). The target words were either a word related to the idiomatic meaning (HAPPY), or a word related to the literal meaning of the last word (SAINT) or an unrelated control word (UMBRELLA). We found that when the sentence can be predicted as idiomatic early on (as in experiment 1), subjects were faster at deciding on the idiomatically related target than either on the literally associated target or on the unrelated target. But when the string didn't call to mind its idiomatic completion until the very end (as in experiment 2), the decision on the target literally related to the last word was faster than on the idiomatic and on the control targets. In a third experiment, the same idioms and targets of exp. 2 were used but the targets were presented 300 ms after the end of the idiom. The results showed that both the idiomatically related target and the literal one were faster than the control word.

The results of the three experiments --at least for the class of idioms considered-- directly contradict both the Lexical Representation Hypothesis and the Direct Access Hypothesis because they fail to show initial activation of the idiomatic meaning. This suggests that idiomatic interpretations need time to be formed along with persistence of literal interpretation. We suggest that idioms are not stored as separate entries in the mental lexicon, but rather that their meaning is associated with particular configurations of words that become available when sufficient information has been received by the listener. The words participating in the configuration are the same words normally accessed during comprehension.

The amount of input necessary to render the configuration recognizable is suggested by the notion that every verbal phrase idiom has a crucial component, a key, that is more relevant than other constituents for detecting the idiom. The idiomatic string cannot be recognized, that is the configuration cannot emerge, before its key has been accessed from the mental lexicon. One claim of this approach is that there is only a single literal path for processing an idiomatic string until, sometime after the activation of its key, the configuration emerges.

The pattern of results found in the three experiments is

explained as follows. When the key occurs early in the string (as in exp.1), only the idiomatic interpretation is found at the very end of the string. When the key is in the last word (as in exp.2), the configuration is not detected until that last word has been accessed, and so only the literal meaning is facilitated. Only after this point can the configuration be identified and the idiomatic meaning can emerge (as showed by exp.3).

This hypothesis has certain advantages, for instance it is able to account not only for the results of the experiments but also for such phenomena as the syntactic flexibility of idiomatic expressions and for different kinds of idioms. However it also faces of course a number of problems. First it has to characterize in a more formal way notions such as configuration and key. Second it has to deal with the well known effect of context in language comprehension.

The experiment we will present was designed to investigate this latter problem, that is, how context can affect the recognition of an idiomatic expression.

METHOD

Subjects. 37 undergraduates volunteered for the experiment, which lasted about 25 minutes. None of them had previously participated in an experiment of this sort.

Materials. The idioms and targets used were the same as those in experiments 2 and 3, above. They had been selected as follows: A group of forty familiar italian idioms having no possible literal counterpart were chosen. In order to be sure that they were not predictable, the idioms were inserted in low informative sentences (e.g., He sent the person to the devil) which were mixed with 175 literal filler sentences (e.g., The man was leaning on the door). Five list were created, each containing a fragment of each sentence in a random order. Their length varied across the lists so that in list 1 the fragments were in the shortest version (e.g., He sent...) and in list 5 in the longest (e.g., He sent the person to the...). Each list was given to 15 subjects who were requested to complete every sentence in a meaningful way. This procedure allowed us to select 12 idiomatic expressions which were completed idiomatically by not more than 5% of the subjects across the five lists.

For each of the 12 idiomatic expressions, a sentence biasing toward the idiomatic meaning was constructed (e.g., After again quarrelling with her boyfriend, she sent him to the devil). Each idiom was paired with three targets to be used as visually presented words during the experiment. The idiomatically related target was obtained by selecting the most frequent word expressing the paraphrase of the idiom given by 15 judges (idiomatic target) (e.g., AWAY); the target related to

the literal meaning of last word of the idiomatic string was obtained asking to a different panel of subjects a word literally associate to it (literal target) (e.g., HORNS); and the third target was an unrelated control (control target) (e.g., TROUT). In order to make sure that the lexical decisions to the twelve idiomatic targets, the twelve literal targets and the twelve controls were comparable in isolation, a pretest was conducted in which the targets were included in a list of 154 words and 154 non-words. The list was presented in four randomized blocks to 20 subjects who performed a lexical decision task on the items. One idiom was discarded because of too long RT to its literal target. The mean reaction times for the idiomatic target words, the literal target words and the control words were 537 ms, 535 ms and 538 ms respectively.

In addition to experimental materials, filler materials were constructed consisting of 60 sentences, not containing idioms, approximately of the same length as the experimental sentences. Half of them were paired with visually presented words and half to visually presented non-words obtained by an orthographical alteration of real words. For six words there was a semantic association with one of the words of the sentences, whereas the remaining were unrelated to the sentences.

The 71 sentences (11 experimental and 60 filler) were recorded, in a random order, on one channel of a tape recorder by a male speaker. On another channel an impulse of 1000 hz, inaudible to subjects, was automatically placed precisely at the offset of the last word of each idiom. For filler sentences, these impulses were placed so as to cover all sentence positions. The impulse caused a word to be displayed on the screen of a micro-computer for 1500 ms, and started a timer which either stopped when the subject pressed the response button (the space bar) or reset automatically after 5 sec.

Each idiom was presented only once, so the three sets of visually presented targets were constructed so that in one set the target paired with an experimental sentence was related to the idiomatic meaning, in another set it was associated to the literal meaning of the last word of the idiom, and in the third it was the control word. The filler targets were the same in the three sets. In each set, there was a total of 11 experimental visually presented targets. Each set was paired with the list of sentences and randomly assigned to an equal number of subjects who acted as their own controls.

Procedure. Subjects were tested individually and sat in a sound-attenuated room in front of an Apple micro-computer screen connected to a tape recorder. They were instructed to perform a lexical decision task on a word that would be displayed on the screen, while listening to a list of sentences. They were instructed to press the space bar with their dominant hand if the letter string was an actual word, and to do nothing otherwise. After ten practice trials, one of the three sets

paired with the list of sentences was presented to each subject. RT to lexical decisions were collected.

Halfway through the experimental session there was a brief interval during which the subject could rest. Subjects were instructed to pay attention to the sentences because at the end of the experimental session they would be asked questions about them. Immediately after the end of the session, subjects were given 20 sentences, each printed on one card. Half of them had been presented during the experiment and half had not, but were derived from sentences actually heard by changing one or more words. Seven subjects who failed to reach a score of 60% of correct recognition were excluded from subsequent analysis.

RESULTS

The mean percentage of errors was .09. In order to reduce extraneous variability, data points plus or minus two standard deviations from the mean RTs of each subject (4.8% of all responses) were excluded from the analysis. The mean RTs of the correct responses for each of the three experimental conditions were: Idiomatic targets 622 ms, literal targets 616 ms and controls 665 ms. The main effect for sentence type was significant both in the analysis by subjects ($F(2,58)=8.11$, $MSe=2646$; $p< .001$) and in the analysis by materials ($F(2,20)=3.80$, $MSe=2871$; $p< .05$). Planned non-orthogonal comparisons showed that idiomatic and literal targets did not differ from each other, and that both were faster than controls (Idiomatic vs Literal: $F(1,58)=0.22$ n.s.; Idiomatic vs Control: $F(1,58)=10.43$ $p< .005$).

DISCUSSION

These findings suggest that context affects the recognition of an idiomatic expression. With neutral sentences, we found that when the idiom key is in the last word of the idiomatic expression, the idiomatic interpretation was not yet formed at the very end of the idiomatic string so only the literal meaning of the key was activated. But when context biased the listener toward the idiomatic interpretation of the sentence, the contextual information acts to facilitate the recognition of the idiomatic expression even though it does not affect access of the literal meaning of the last word in the idiom. The fact that we didn't find a significant difference between the idiomatic and the literal targets does not, of course, imply that they are activated at the same time. Further evidence seems necessary for assessing the time course of the literal and idiomatic interpretation when context orients the listener toward the idiomatic meaning of the sentence.

The Configuration Hypothesis seems to be consistent with these results. It is reasonable in fact to suppose that a

configuration can be pre-activated by biasing contextual information. Thus, the context is effective in speeding up the activation of the idiomatic configuration which otherwise needs time to emerge, as demonstrated by earlier experiments.

Methodologically, the different pattern of results obtained in earlier experiments and in the present one using the same paradigm, and either identical or comparable materials, both in and out of context, suggests that the lexical decision task is indeed sensitive to immediate, perceptual processes without being susceptible to backward priming, at least in sentential contexts.

Theoretically, the present findings cast further doubts on the Direct Access Hypothesis proposed by Gibbs (1980, 1986) since this hypothesis predicts that people will directly access the idiomatic interpretation of a sentence irrespective of its literal analysis or context. Clearly, this was not the case here.

REFERENCES

Cacciari, C. & Tabossi, P. (1987). The comprehension of idioms. Manuscript submitted for publication.

Estill, R. & Kemper, S. (1982). Interpreting idioms. Journal of Psycholinguistic Research, 6, 559-568.

Gibbs, W. R. (1980). Spilling the beans on understanding and memory for idioms in context. Memory and Cognition, 8, 149-156.

Gibbs, W. R. (1986). Skating on thin ice: Literal meaning and understanding idioms in conversation. Discourse Processes, 9.

Glass, A. (1983). The comprehension of idioms. Journal of Psycholinguistic Research, 12, 429-442.

Katz, J. (1973). Compositionality, idiomaticity and lexical substitution. In S. R. Anderson & P. Kiparsky (Eds.). A Festschrift for Morris Halle (pp.357-376). New York: Holt, Rinehart & Winston.

Ortony, A., Schallert, D., Reynolds, R. & Antos, S. (1978). Interpreting metaphors and idioms: Some effects of context on comprehension. Journal of Verbal Learning and Verbal Behavior, 17, 465-478.

Swinney, D.A. & Cutler, A. (1979). The access and processing of idiomatic expressions. Journal of Verbal Learning and Verbal Behavior, 18, 523-534.

ACKNOWLEDGMENT

Preparation of this study was supported by grant No.8600499 from the Fondi 60% Ministero Pubblica Istruzione and by Fondi 40%. We are grateful to Linda Laghi for her assistance in

collecting data and to Corrado Cavallero for his help in analyzing data.

Action Planning: Routine Computing Tasks

Suzanne M. Mannes & Walter Kintsch

University of Colorado, Boulder

The tasks we are concerned with here are routine operations involving a file and mail system. Our subjects are experienced computer users, for whom such tasks present no challenge. However, solutions for these tasks cannot be precompiled, because while the elements of each task may be familiar, and indeed overlearned, they are often put together in novel sequences. In other words, we are not dealing with fixed scripts which only need to be retrieved from memory, but with plans for action that are generated apparently effortlessly in the context of each specific task. The question addressed here is how these plans are generated.

THEORETICAL BACKGROUND

The problem statement for each task is a piece of discourse, usually a sentence or two with which the experimenter instructs the subject what to do. (More generally, one can assume that computer users instruct themselves in similar way in the course of their activities). The theory of discourse understanding, developed in a quite different context by Kintsch & van Dijk (1978), van Dijk & Kintsch (1983), and Kintsch (1988), will be used to model the way in which subjects comprehend this discourse.

We are interested in how the instructional text activates the knowledge a subject has about the tasks involved, resulting in a well-formulated plan for action. We are not studying the actions themselves: this is probably best done within a GOMS- or CCT-like framework (Card, Moran, & Newell, 1983, Kieras & Polson, 1985). What we do here is generate the GOAL statement, which is required to activate the high-level productions in these models. Once you have a PLAN – in the sense discussed below – you can then activate a sequence of productions for executing that plan, in the manner described by these models.

Understanding an instructional text means coming up with an appropriate plan for action. The approach we take to modelling this process is analogous to our previous work on word arithmetic (Kintsch & Greeno, 1985). Understanding a word arithmetic problem means generating a mental representation of the sets and set relations involved, on the basis of which the correct computations can be performed. This is achieved by activating both relevant general world knowledge and specific knowledge about arithmetic. Analogously, in the present case, we need to activate knowledge about the general nature of the task we want to perform, as well as knowledge about the specific computer system on which we are working. Thus, the same general theory of comprehension is being applied to two different domains.

PROTOCOL ANALYSES

Three experienced computer users were given the following three tasks with standard think-aloud instructions:

(a) *Include an address that you know into a file on the computer.*

(b) *You got a manuscript from someone and you want to edit it, making modifications, and then return it to the person you got it from.*

(c) *You got a message from someone while in mail asking for a paragraph out of a manuscript of yours to be sent to them.*

Their actions as well as their verbal protocols were recorded and analyzed by means of a method described in detail in Kintsch & Mannes (1987). The resulting temporal sequences of action planning statements and elaborations provided the starting point for our analyses. They gave us

some idea about the sort of knowledge that is activated when subjects perform these tasks, as well as about the sequence of planned and actual actions involved. Table 1 shows a greatly abbreviated summary of the verbalizations and actions of Subject 1 performing Task b.

We note the evidence for backward reasoning in Table 1, which is typical for these protocols. It is the anticipation of things to come which often drives current actions. In Table 1, the subject recognizes that in order to be able eventually to send the revised file, she must first enter the mail system to get the file[1†]. Our theoretical goal will be to simulate the sequence of planned and actual actions, as observed in our data (but not the accompanying verbalizations).

THE MODEL

The main components of the construction - integration model proposed by Kintsch (1988) are a propositional textbase derived from the instructional text, an associative long-term memory, a knowledge activation process by means of which the textbase propositions activate information from the long-term memory, and, following these construction phases, an integration process which selects what fits together, and deactivates what is contextually irrelevant. Furthermore, these processes are cyclic, because the text is processed in sentence-like units rather than as a whole, and because when some action is performed, the state of the world changes. Thus, processing is repeated until the desired change in the state of the world has been achieved, or until it fails.

Long-term Memory

Forty-two propositions were taken from the protocols of subjects performing the three tasks described above to simulate a portion of a user's long-term memory network. Interconnections in this net were computed by finding all the instances where a given proposition shared an argument with another proposition, and where it was embedded in another proposition. Additive connection values of .5 were used for each case of argument overlap or embedding. A LISP program computes the interconnection values among all the items in long-term memory, creating a matrix of size n x n, called the Connectivity Matrix. This matrix approximates an association matrix: since we do not know the actual associative strengths among the propositions in long-term memory, we use this crude and fallible, but relatively simple and objective method to estimate these values. The second portion of the simulated long-term memory consists of 15 PLANS, as shown in Table 2.

Each plan has three fields, a plan name, preconditions, and outcomes. Preconditions correspond to propositions designating states of the world which must be fulfilled in the environment for the plan to be executable. Outcomes correspond to propositions which become

Table 1. Abbreviated Protocol: Subject 1, Task b (REVISE).

GET INTO MAIL	TO GET TO THE FILE
SAVE MESSAGE TO FILE	ASKS FOR FILE NAME
EXIT MAIL	
EDIT FILE	I'D BE IN EMACS WITH MANUSCRIPT FILE
EXIT EDITOR	
SEND MAIL MESSAGE	ASKS WHO TO SEND TO

1† The verbal protocols provide other instances of this reasoning, e.g.:
First I'd send my method section of the paper. I'd get into the editor one way or another and – first I'd have to look at the paper you are talking about. So I would have to think through how I stored that in my directory structure... So I'd have to think through how I have organized my stuff and go find the paper and then bring it up in the editor. I'd read your letter and then I wouldn't want to reply right away. I'd want to go up – you know – figure out what was going on.

states of the environment as a consequence of executing a plan. Plan names are propositions, thus consisting of a predicate and a number of arguments. Sets of plans, called fields, may be mutually exclusive (e.g. the four ways to send mail in Table 2). In such fields, the most strongly activated plan inhibits the other members.

Connections between the propositions in the long-term memory net and the plans are computed by the same algorithm based on argument overlap and embeddings, based upon the plan name only. An exception must be made, however, for certain types of proposition which we call REQUESTS and OUTCOMES.

Requests are the imperative verbs that are used in problem statements (requesting the user to do something - edit, send, etc). The relationships between these verbs and plans has to be more precise than the one afforded by argument overlap: each particular request must be associated directly with an appropriate plan. Thus, the REQUEST *EDIT* is tied to the plan (EDIT FILE) by a value of +1, and has 0 connection strengths with all other plans. Each OUTCOME proposition is similarly connected by a value of +1 to plans that produce this outcome, by a value of -1 to plans which produce an incompatible outcome but which include task relevant objects, and by a value of 0 to plans which produce irrelevant outcomes (because they deal with objects that are not involved in the task at hand). Only ultimate goals are connected in this way: other plans which need to fire before the ultimate plan can be accomplished must be contextually activated through the network as a whole.

TEXTBASES AND SITUATION MODELS

From the text of a problem (such as the three tasks mentioned above) a propositional textbase is derived by handcoding, in the manner of Kintsch (1985). Each text proposition then activates two other propositions which are associated with it in the long-term memory net. This is done by computing the argument overlap and embedding relations between that proposition and all other propositions in long-term memory and normalizing this vector (so that the sum of all interconnections is 1).The resulting connection strengths values can then be treated as probabilities,and two associates of each propositions can be sampled with replacement.

Table 2. List of PLANS.

Plan Name	Preconditions	Outcomes
(COPY TEXT FILE)	(@SYS) (IN FL DIR)	(IN TXT FL) (IN FL DIR)
(COPY FILE)	(@SYS) (IN FL DIR)	(IN FL DIR)
(READ FILE)	(@SYS) (IN FL DIR)	(READ FL)
(EDIT FILE)	(@SYS) (IN TXT FL) (IN FL DIR)	(IN TXT FL) (IN FL DIR)
(FIND FILE)	(@SYSTEM)	(IN FL DIR)
(SEND TXT FL ML)	(@ML) (IN TXT FL) (IN FL DIR)(IN MSG ML)	(RECV TXT)
(SEND TXT MAIL)	(@ML) (IN MSG ML)	(RECV TXT)
(SEND TXT FL SYS)	(@SYS) (IN TXT FL) (IN FL DIR)	(RECV TXT)
(SEND TXT SYS)	(@SYS)	(RECV TXT)
(COPY MESSAGE)	(@ML) (IN MSG ML)	(IN TXT FL) (IN FL DIR)
(READ MESSAGE)	(@ML) (IN MSG ML)	(READ MSG)
(FIND MESSAGE)	(@ML)	(IN MSG ML)
(SAVE MESSAGE)	(@ML) (IN MSG ML)	(IN TXT FL) (IN FL DIR)
(ENTER SYSTEM)	(@MAIL)	(@SYS)
(ENTER MAIL)	(@SYSTEM)	(@ML)

Once again, request propositions require special treatment: they do not randomly activate associated knowledge, but must be used to retrieve an expected outcome. Specifically, a proposition of the form (REQUEST X (Y)) together with (OUTCOME $) is used as a joint retrieval cue to retrieve W(Z) - the outcome of doing X(Y) which is associated to both retrieval cues, as in Raaijmakers & Shiffrin (1981) and Kintsch & Mannes (1987). Thus, we now have text propositions, plus their associates and outcomes. To these we add all 15 plans: when trying to solve a problem of the type considered here, all plan-knowledge must at least have the potential to be activated.

The interconnections among the nodes in the resulting matrix are computed in the same way as in the original long-term memory matrix. There are, however, two further complications, concerning the relations between outcomes and plans, and among the plans themselves. If all of the outcome propositions for a plan already exist in the models current state, they inhibit the plan that produces these outcomes. Thus, if the location of FILE-X is already known, the plan (FIND FILE-X) is inhibited. Expected outcomes of plans, on the other hand, are connected to other plans in the same way as in the long-term memory matrix, i.e. they activate plans that produce these outcomes, and inhibit plans that produce incompatible outcomes. Plans themselves are interconnected by a causal chaining mechanism. That is, a plan that requires precondition X activates all plans that have X as an outcome. Thus, the interconnections among plans represent the user's knowledge about the causal relations in the system.

In this manner, a connectivity matrix of size n x n is obtained for each text, corresponding to the original m text propositions and to the (n-m) associates, outcomes, and plans that have been added to the text. An initial activation vector with n elements is then constructed, with activation values 1/m for the m original text propositions and 0 for everything else. This vector is then repeatedly postmultiplied with the connectivity matrix, and renormalized after each multiplication, to compute the spread of activation among the elements of the vector. When the activation vector reaches an arbitrary criterion, such as an average change after a multiplication of less than .0001, the activation pattern is considered to reflect the stable situation model formed by the user, given the particular task and the knowledge assumed here.

In terms of the van Dijk & Kintsch (1983) model, we have started with a small *textbase*, which activated both relevant and irrelevant *knowledge*. We then used an *integration* process to deactivate the irrelevant knowledge that had been introduced, thus arriving at a *situation model* that corresponds to the user's understanding of the situation and the task to be performed.

Plans and Action

Plans end up at different levels of activation in the situation model. We postulate an executive process that checks the plan with the highest activation level and executes it if its preconditions are satisfied. If they are not, it goes to the next highly activated plan, and so on. If the plan executed has as its outcome the expected outcome (which was retrieved from the long-term memory by means of the original request), the problem is solved. If not, the consequences of the plan that was executed are added to the situation model, and the integration process is restarted, resulting in a new pattern of plan activation. This process is repeated until the desired outcome results (or is stopped when it goes hopelessly awry).

THREE EXAMPLES

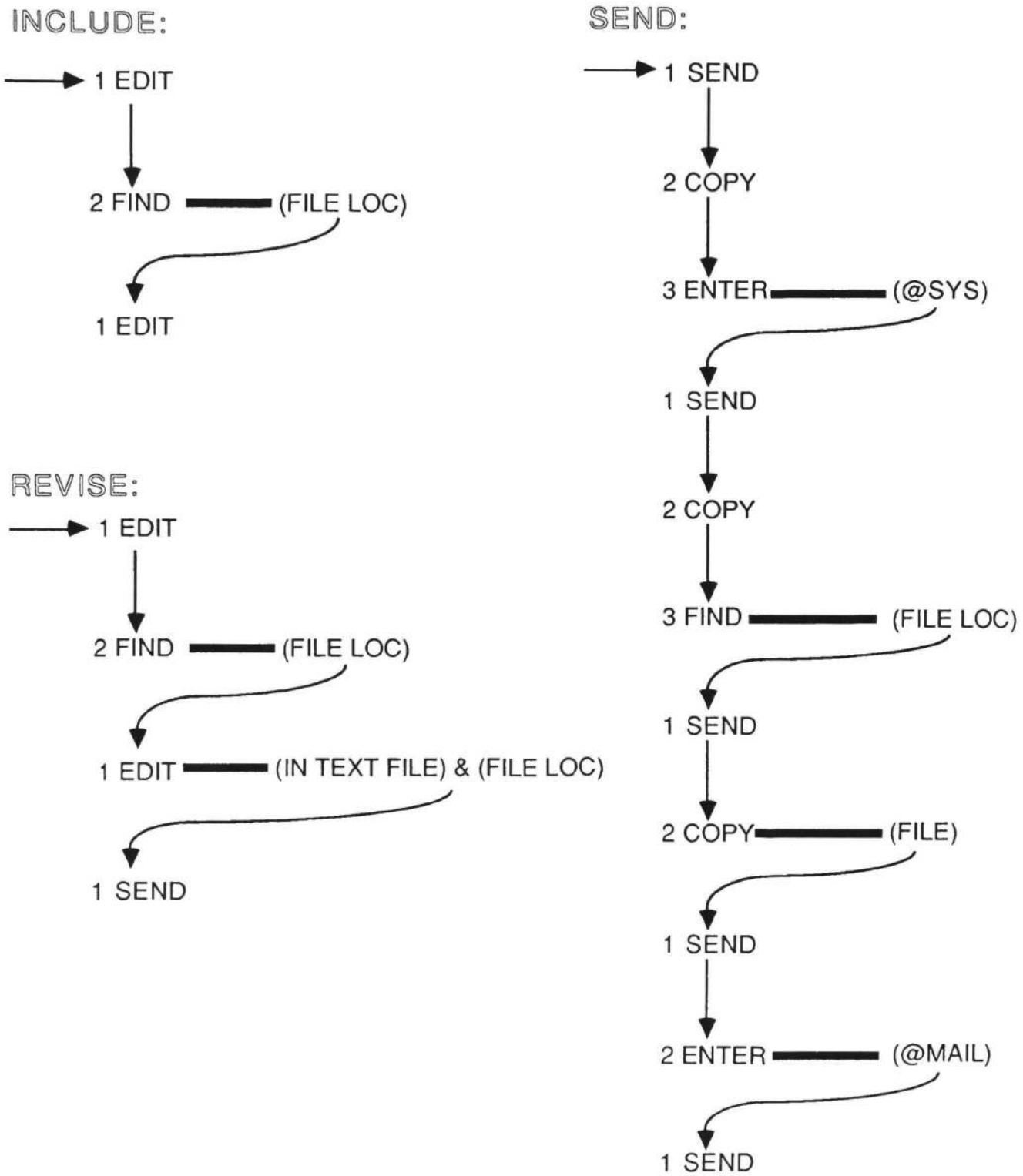
The results of a simulation with this model of the three problems shown above are partially depicted in Table 3. Only the plans actually considered by the executive process are shown, together with rankings indicating their relative activation values. For the *INCLUDE* problem, the (EDIT FILE-LETTER) plan receives the highest activation value in the first round. It cannot be executed because one of its preconditions is not fulfilled: the location of FILE-LETTER is unknown. Therefore, the second-ranked plan is considered. This happens to be (FIND FILE-LETTER), it is executed, and results in the addition of the proposition (KNOW FILE-LETTER LOCATION) to the situation model. After a new integration phase, the (EDIT FILE-LETTER) plan again is the most highly activated one. It is now possible to execute this plan, which provides

the desired outcome (IN TEXT-ADDRESS FILE-LETTER), completing the problem. The *REVISE* example is segmented into two components by the linguistic cue *and then*: such temporal connectives tend to be used as signals of episode boundaries, and hence as segmentation cues (Kintsch & Mannes, 1987). The request which is acted upon first is to *revise the manuscript*: the model wants to EDIT, but cannot do it, so its executes the next highest plan, which results in a new proposition characterizing the state of the system, (KNOW FILE-MANUSCRIPT LOCATION). After a new integration process, EDIT is still at the top, but now it is possible to execute this plan. As a result, the propositions (IN TEXT-MANUSCRIPT FILE-MANUSCRIPT) & (KNOW FILE-MANUSCRIPT LOCATION) get added to the textbase, the request to revise is dropped, and the request to send it is substituted. The integration process selects the plan (SEND TEXT-MANUSCRIPT FILE-MANUSCRIPT @SYSTEM), all of its preconditions are met, the plan is executed and the problem is solved. The *SEND* problem is somewhat more complex and requires five construction-integration cycles before the solution is arrived at. From the very beginning, the model wants to SEND, but is unable to do so. All it can do in the first cycle is to leave the MAIL system. In the next cycle, it manages to locate the file it needs, so that it can copy the to be sent text into a FILE-TEXT. However, in order to send this file, it wants first to re-enter the mail system, because it prefers to reply to the message it received, rather than sending a new one as in *REVISE*.

CASE-BASED REASONING

In the simulations described above, the program used only its general knowledge about plans in arriving at solutions. What would happen, however, if these problems, or similar ones, were presented for a second time? Although we don't have any direct empirical evidence as yet, we would expect to see examples of case-based reasoning. It appears plausible that the to-be-comprehended text would remind subjects of having solved this problem, or a similar one, before. There are many ways in which a case-based reasoning capability could be added to the present simulation. As an initial exploration, we decided to make some simple assumptions about memory for an episode. Specifically, we assumed that what would be remembered from an episode would be the three most highly activated text propositions, together with the plans that were executed in performing the task. Thus, four cases, each consisting of the three propositions and the corresponding plans, were added to our long-term memory. (There were four cases, because the Revise problem consisted of two episodes). The three problems were then re-run. In each instance, the text propositions activated the appropriate earlier case, and sometimes even other cases. Thus, the Revise text reminded the system not only of the earlier Revise episodes, but also of the Send episode, which also involved a text and file called *manuscript*. In the comprehension network, each case was connected only to the plans that were actually executed in this episode. Not surprisingly, this had the effect of further strengthening these plans, and all problems were solved correctly, as before. Substantial portions of the activation of each plan, however, derived from its connection to earlier episodes: between 40% and 52% of the total activation of non-ultimate plans (i.e. plans that had to be executed before the ultimate goal could be achieved) was acquired from remindings of previous cases. For ultimate plans, this proportion was lower, ranging between 20% and 29%. Thus, cases helped particularly in the early part of the comprehension process, while the last step is well defined by the text anyway.

Table 3. Traces of solutions to the INCLUDE, REVISE, and SEND tasks.



CONCLUSIONS AND FURTHER DEVELOPMENTS

The model's ability to simulate the comprehension processes for the three tasks considered here is promising, though of course hardly conclusive. We need to expand the size of our long-term memory, the number of plan alternatives, and the range of problems considered, and see how the present model will perform. We need to explore more systematically than we have done so far the parameter values used in this simulation. (Basically, we chose the first combination of values that worked for the present simulation). We need to explore various ways of making our simulation more elegant and efficient (e.g. it may not be necessary to introduce a separate executive component to check whether plans are executable, etc.). Equally importantly, we need to find a way in which the details of the model predictions can be evaluated against empirical protocols. All we have done so far, is to note a generally satisfactory overall correspondence.

In spite of the fact that this work is as yet quite incomplete, our progress so far warrants some optimism. From the standpoint of problem solving work, what we have done is not spectacular: we do a means-end, backward problem solving that probably would not have challenged the General Problem Solver. However, we do this within a framework of a general theory of discourse comprehension. We are not proposing a model of problem solving in the computing domain *de novo* but merely adapt a comprehension model to this domain by representing the domain knowledge as plans. Should we be able to work out and test more fully the model sketched here, we plan to use the model to explore the question how novice users who do not have the kind of knowledge upon which behavior is based in our simulations can be helped to perform routine computing tasks.

REFERENCES

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Erlbaum.

Kieras, D. E., and Polson, P. G. (1985). An approach to the formal analysis of user complexity. *International Journal of Man-Machine Studies*, **22**, 365-394.

Kintsch, W. (1985). Text processing: A psychological model. In T. A. van Dijk (Ed.), *Handbook of Discourse Analysis*, Vol. 2 (pp. 231-244). London: Academic Press.

Kintsch, W. (1988). The use of knowledge in discourse processing: A construction-integration model. *Psychological Review*, in press.

Kintsch, W. & van Dijk, T. A. (1978). Toward a model of text comprehension and production. *Psychological Review*, **85**, 363–394.

Kintsch, W. & Greeno, J. G. (1985). Understanding and solving word arithmetic problems. *Psychological Review*, **92**, 109-129.

Kintsch, W. & Mannes, S. M. (1987). Generating scripts from memory. In E. vanderMeer & J. Hoffmann (Eds.), *Knowledge aided information processing* (pp. 61-80). Amsterdam: North-Holland.

Raaijmakers, J. G. & Shiffrin, R. M. (1981). Search of associative memory. *Psychological Review*, **88**, 93-134.

van Dijk, T. A. & Kintsch, W. (1983). *Strategies of discourse comprehension*. New York: Academic Press.

Using Conversation MOPs to Integrate Intention and Convention in Natural Language Processing*

Elise H. Turner and Richard E. Cullingford
School of Information and Computer Science
Georgia Institute of Technology

Natural language processing systems must consider both convention and intention since both are a part of language. The conventions allow a society of language users to agree on many different facets of the language, from word meanings to discourse structure. Although it is easy to imagine a situation where convention almost completely governs a conversational exchange, a speaker often manipulates the conventions to help achieve some goal. When this happens the hearer must be able to understand the intention of the speaker to be able to fully understand the utterance.

Most previous work in natural language processing has focused on either intention or convention. Finding conventions and using them in computer system interfaces has been the focus of a great deal of research (for example, Grosz, 1977; McKeown,1985; Reichman, 1985; Sidner, 1983). In this work high level discourse structures are isolated and automated so that conversational flow appears natural.

Other research in natural language processing has focused on the intentional component of language. Philosophers of language recognized that each utterance is an act and, as such, can be subject to the same sorts of successes and failures as any other action (Austin, 1962; Searle, 1969). This insight led computer scientists to study speech acts in terms of planning (Allen & Perrault, 1980; Carberry, 1986; Cohen & Perrault, 1979; Litman, 1986). In their work, speech acts function as operators in plans. Conditions that must hold for the speech act to be used and the effects of the speech acts are associated with each operator. This method has been extended to handle difficult discourse such as indirect speech acts, where the literal meaning of the speech act is not intended by the speaker, and ellipsis, where a part of the utterance must be inferred. Planning methods provide flexibility by allowing the system to reason deeply about the intentions of the speaker and to form new dialogues from actions which can be put together in any way that follows the plans of the speaker and hearer.

Planning methods are not cognitively plausible and may expend computational effort unnecessarily because they do not take advantage of the conventions in language. Carberry and Litman incorporate discourse structures into their systems, but both take a plan-based approach. Our approach follows convention, whenever possible, as a shortcut in planning. However, the conventions are represented in such a way that they are flexible. The conventions can also be used as plans for inferring the user's plan or finding an action for the system if the need for that information warrants the additional computation.

We are implementing our ideas in JUDIS (JUlia's DIscourse System), a system which provides a natural language interface for Julia (Cullingford & Kolodner, 1986). Julia is a highly interactive advice giving system using a variety of problem solvers to provide assistance in common sense domains. The current version of Julia functions as a caterer's assistant helping the user to plan a meal. In the initial implementation of JUDIS, we have focused on the beginning portion of conversation, called the *initation phase* (Douglas, 1984), to narrow considerations to those which are directly affected by a fairly conventional, yet potentially flexible, conversational structure.

MOPS AND CONVERSATION MOPS

In our view, conversational conventions are learned from many past experiences with conver-

*This research is supported by the National Science Foundation, grant number IST-8608632. A longer version of this paper has been submitted to *Discourse Processes*

sation. These conventions can be used to achieve the same goals that have been associated with them in the past. However, since no conversation is expected to exactly repeat the past, the representation for conventions must allow general rules to be instantiated for a specific situation and must be flexible enough to allow the intentions of the conversants to impact them.

To represent these conventions, we use MOPs, or *memory organization packets* (Schank, 1982). A MOP is a schematic structure used to organize long-term, conceptual episodic memory. Each MOP represents a generalized episode which satisfies a goal. The building blocks of these episodes are *scenes*. Scenes represent pieces of the episodes and are associated with instrumental goals (Schank & Abelson, 1977) that are used to achieve the higher level goal of the MOP. For example, a MOP to represent buying theater tickets could include such scenes as: GET-MONEY, GO-TO-THEATER, GO-TO-BOX-OFFICE, ASK-TICKET-AVAILABILITY, SELECT-SEATS, PAY-FOR-TICKETS, and CHECK-TICKETS. The goals associated with the scenes and MOPs are important for capturing the intention in a discourse fragment.

MOPs can be stored in and retrieved from a *dynamic memory* (Kolodner, 1984; Schank, 1982). A dynamic memory stores MOPs using indices based on their important features and creates new MOPs from specific events that share features. In current implementations, the programmer decides which features will be important, although this could theoretically be learned by the memory. A specific value of an important feature is called a *predictive index*. These are used to find information in the memory. When a new episode is to be stored, a place for it is located in memory using the same indices that will be used to find it. If some predetermined number of episodes are found in this location, a generalized episode is created and the specific episodes are indexed under the MOP by their important differences from the MOP. If the location in which to store an episode already contains a MOP, the episode is indexed under this generalized MOP.

There is evidence that *conversation MOPs* represent conversational structures in human beings (Kellermann, Broetzmann, Lim, and Kitao, in press). Kellermann et al. studied the MOPs shared by members of a society, focusing on conversation MOPs that represent informal initial conversations such as those that occur between two people meeting in a grocery store or at a party.

The conversation MOPs that are used in JUDIS reflect these shared MOPs.[1] JUDIS currently has conversation MOPs representing general rules of conversation, such as turn-taking and question answering, as well as rules for the expected structure of a complete conversation. A complete conversational system will have very general MOPs for conversational goals that arise in most conversations, and more specific MOPs to capture conventions from particular contexts. Since many goals may be active in one conversation and additional MOPs will be chosen to guide the conversation in the given context, many MOPs will apply at once. JUDIS does not construct a complete plan from the conversation MOPs, but follows a MOP when an utterance or a goal of the system brings the MOP to the attention of the planner. (Details of this process are discussed below.) This method allows the system to respond to user utterances that do not fit into the MOP that the system is following. The method demands a mechanism for conversational control that can keep track of the many active conversation MOPs.

Since Julia is a caterer, JUDIS has a caterer's conversation MOP which gives the topics expected in a conversation between a client and a caterer. The caterer's MOP has a scene for the initiation phase. We have adapted the initiation phase found by Kellermann et al. (in press) to apply to a computerized caterer's assistant.

A simplified version of a conversation MOP for the initiation phase between a computer caterer and a new client appears in Figure 1. The initiation phase MOP contains the characters that appear in this scene. JUDIS assumes that it is functioning in the role of caterer, but it could use this MOP if it were the customer of a caterer. The goal of the MOP is simply to start a conventional conversation. This goal could be in service of some other goal such as getting to know someone, getting information, or, in the case of Julia, being a good and polite businessperson.

The scenes in the initiation phase give the actions that usually take place in this phase of the conversation. The scenes in the initiation phase are fairly well-defined; in later portions of the dialogue, scenes may be more vague. For specific

[1] JUDIS does not currently learn conversation MOPs, although this is a long-term goal of this research. We believe that the MOPs the system has been given are potentially the products of generalization in a dynamic memory.

characters: caterer, customer
goal: conventional-conversation
scenes:
  greeting
    The user and Julia exchange greetings such as "Hello." or "Hi."
  introduction
    Julia introduces itself; the user may introduce him- or herself if the user id given at login is not the user's name
  positive-evaluation
    exchange "It's nice to meet you." or some other positive evaluation of the introduction.
  ask-health
    both can ask "How are you?"
reason-for-presence
    Julia asks the reason for the person's visit
seq-of-events: greeting, introduction, positive-evaluation, ask-health, reason-for-presence
mandatory-scenes: greetings, introduction, reason-for-presence

Figure 1: The Initiation Phase MOP

rules of conversation, scenes may be very general actions, such as forming a question. Here, information from the current context will be used to fill in specifics of the question at the time the question is asked. The order of the scenes is given by seq-of-events. In this initiation phase MOP the scenes are totally ordered, but our representation also allows partial orderings. The scenes listed as mandatory scenes are expected to occur in this phase simply because of convention, not due directly to the active goals of the system.

Scenes which are not mandatory but which are present in the MOP are considered *optional scenes*. The system initiates an optional scene if it wants to achieve the associated goal. If the user initiates an optional scene, the system can infer that the user had the associated goal. Optional scenes give the conventions some flexibility by giving the system a conventional way to solve a goal, so it does not have to do extensive planning, but not forcing execution if the associated goal does not need to be achieved.

Later portions of the conversations with Julia will be controlled more by the system's and the user's problem solving. The goals associated with conversation MOPs can account for intention in conversation, and the use of MOPs in the problem solving systems of Julia allows the problem solving plans of the user to be available to JUDIS. The caterer's conversation MOP allows JUDIS to have reasonable expectations of topics to be covered during the entire course of conversation, without expending a great deal of computational effort. This overview allows JUDIS to integrate questions and information from Julia's different problem solvers. If an utterance requested by one of Julia's subsystems does not relate to the current topic and the utterance is not too important to the problem solver, the system can check to see when the topic will be discussed and include the utterance at the appropriate time.

AN OVERVIEW OF JUDIS

The current focus of JUDIS is on using conversation MOPs in the initiation phase. This phase is guided almost exclusively by its conversation MOP. The conversants are concerned with beginning the conversation, not with any problem that may be discussed later. This limits the initiation phase to goals that are related to conversation and not problem solving. The goals associated with the optional scenes will be persistent goals, like *polite-conversation*, that will set the stage for the rest of the conversation. To infer the user's plan we do not need to look at possible problem solving strategies. The system can also ignore relating problem solving to its own planning at this stage. Although we do not consider highly irregular dialogues, the initiation phase does contain

optional scenes which allow us to study several aspects of the system's potential flexibility. The initiation phase also forces the system to handle several MOPs which account for the same utterance.

Finding the Initial MOPs

Initially, JUDIS must choose which MOPs are pertinent to the conversation. It starts with the goal of being a good caterer which includes the goals for carrying out a coherent conversation. The context of being in the catering environment is used to select the caterer's conversation MOP from memory. The familiarity of the caterer with the client is used as an index into memory to select the MOP for the initiation phase.

Other conversation MOPs are also selected from the memory at this time. The MOP for turn-taking is selected because of the goal to have a conventional conversation. This MOP is used in so many contexts and has become so general that it is selected whenever the system participates in a conversation.

Serendipitous Goal Satisfaction

There are three different levels of activation that can be associated with a MOP. A *dormant* MOP is in memory and is available to be activated by the conversation system, but is not associated with any active goal of the system or does not apply in the current context. In the case of a new user logging into Julia for the first time, the initiation MOP for familiar users would remain dormant throughout the session. The question–answer MOP is dormant at the start of the dialogue because it is not needed to satisfy any goals. An *active* MOP is associated with a goal that is active, but the MOP is not currently executing. If JUDIS has a question that it is waiting to fit into the conversation, the question–answer MOP for that question is active. MOPs are also active if they are the scenes of active MOPs. For example, while the initiation MOP is executing, the later scenes of the caterer's conversation MOP are active by virtue of being part of the active caterer's MOP. The initiation MOP is considered to be an *executing* MOP because it represents a plan that is currently being followed.

When an executable act or episode being performed to satisfy a scene in one MOP also satisfies a scene in another MOP, we refer to this as *serendipitous goal satisfaction.* We do not seek to minimize the number of actions executed by finding serendipitous goal satisfaction because we try to limit the amount of computing, not the number of generations, performed to achieve the conversational goal. We believe this method more closely reflects human processing. Very often serendipitous goal satisfaction occurs as part of executing a MOP and does not need to be coerced. For example, a question–answer sequence always satisfies turn-taking.

It is important to recognize when serendipitous goal satisfaction has occurred because it allows JUDIS to avoid repeating plan steps (usually utterances) unnecessarily. It is also important because serendipitous goal satisfaction can change the flow of conversation. To find serendipitous goal satisfaction, whenever an utterance is made, JUDIS checks the active MOPs to see if the utterance can be used to execute a scene in one of them. If it can be, that MOP is seen as executing serendipitously.

JUDIS currently only considers active MOPs when finding serendipitous goal satisfaction. We adopt this strategy to help the planner to maintain its focus. When serendipitous goal satisfaction occurs as the result of an action by the system, if a MOP were important to the goals of the system it would already be active. If the action forced the conversation into following a dormant MOP that was partially satisfied by the action, the system would find the MOP while trying to understand the user's utterance. If the user's utterance executed a scene that was dormant, JUDIS would execute the currently active MOPs, if it could, to respond to the utterance. However, as we work with more complicated portions of dialogue, we may find that we need to examine dormant MOPs more carefully.

Selecting a MOP for Execution

When serendipitous goal satisfaction occurs, more than one MOP is executing. The system cannot simply follow the MOP that it originally chose to execute, since it is not the case that following the originally chosen MOP will guarantee the continued satisfaction of the MOP that is executing due to serendipitous goal satisfaction. Sometimes the way the original goal is achieved demands a response from outside of the chosen MOP. If one person wanted to get the attention of another, he or she could say "Hello" or

"What's up?". "Hello" leaves room for a wide variety of responses available in the initiation MOP, but "What's up?" throws the dialogue into the question–answer MOP and demands an answer. The plan for conversation can also be disrupted by the user. Consequently, the system must choose which scene to execute next.

The choice of the next scene to execute is affected by three factors. As is usual in planning systems, goals are given different priorities and the higher the priority of the goal, the more likely a plan for the goal will be chosen for execution. This is especially important in later phases of the dialogue when an urgent need of problem solving may require the dialogue to be disrupted. The priority of a goal is usually set by its function in the problem solving task, although goals concerned only with maintaining the conversation have a high enough priority to keep the dialogue coherent. Since many of the most general rules of conversation are satisfied in the course of applying other conversation MOPs, such as satisfying turn–taking while executing the question–answer MOP, conversation goals such as *polite-conversation* are given a low priority so that the more specific MOPs have a chance to be executed.

Conversation MOPs are powerful because they can give more guidance than just associating a plan with a goal. The other two factors affecting MOP selection concern the relationship between scenes. The strength of the sequencing, whether or not the executed scene must be immediately followed by the scene being considered for execution, affects the choice of the next scene. If a question has been asked, it is important to answer it with the next utterance. On the other hand, if the appetizer scene has just been discussed, the main course does not have to follow immediately. Another factor to consider in selecting a scene is whether the scene being considered is mandatory or optional. It is important to execute mandatory scenes before optional ones to decrease the likelihood of the dialogue being terminated while mandatory MOPs are left unexecuted.

We use a simple activation model for scene selection. Each of the three factors above contributes to the activation of a MOP. The activation contributed by the goal remains proportional to the priority of the goal. The priority of the goal may increase or decrease, and when this happens the activation that it contributes to the scene will increase or decrease. As long as a goal's priority stays constant, the amount that it contributes to the scene's activation remains constant. However, the activation contributed by position and whether or not a scene is mandatory in the MOP is allowed to decay. This happens because as the time from one utterance to the next utterance in the MOP increases, executing the next scene in the MOP becomes less important. Sometimes the distance is so great that a topic shift must be executed in order to execute the scene.

For example, suppose one conversant asked two questions in a row without allowing the respondent to answer the first. The other conversant must decide which question to answer. Suppose he or she answered the first question first and this led to a discussion of the answer. The longer the discussion continued, the less likely it would be that the second question would be answered. This loss of interest is reflected by the decay of activation in our system.

The activation model also has implications for processing optional scenes. Since optional scenes need stronger activation from the goal than mandatory scenes do, for the system to execute an optional scene the associated goal for the scene must be active.

Two Examples of Initiation Phases

An example of an initiation dialogue that could occur between a user (U) and Julia (J) appears in Figure 2. From the login, Julia knows that this is a new user. The user says "hello" as part of the exchange greeting scene. This also counts as a turn for the user, so the turn-taking MOP is executing as a result of serendipitous goal satisfaction. The scene of returning the "hello" must follow the user's "hello", so JUDIS responds with "hello". The next scene in the initiation MOP is the introduction, so JUDIS has Julia introduce itself. There are no mandatory scenes in the initiation scene and no scene must follow the introduction, so JUDIS waits for the user to take his turn. Since the user must execute this scene, the system can only wait. The user says "How are you today?" which increases the strength of the conversational goal of *polite-conversation*. The "ask-health" scene contains an answer to "how are you?" built into its MOP. When someone says "how are you?" to a person he or she has never met, the hearer does not actually answer the question, but always says something positive. JUDIS knows how to respond from the initiation MOP

U1: Hello.
J1: Hello.
J2: My name is Julia.
U2: How are you today?
J3: Fine.
J4: How are you?
U3: Fine.
J5: May I help you plan a meal today?

Figure 2: Example Initiation Phase 1

and does not need to find the question–answer MOP. This scene continues with the system asking the user's health. The next scene to be selected for execution is reason-for-presence scene. There are no other scenes in the initiation phase, so this scene is chosen. It matches the caterer's goal of catering and can now be executed since mandatory scenes are not taking precedence.

In the first example, the user had the goal of being polite at a high enough priority that he executed the optional goal. But if the user did not have this goal, utterances U2–U3 would be left out of the dialogue. The new dialogue would be executed in the same way as the previous example for the first two utterances. As before, both the initiation MOP and the turn-taking MOP are executing. The initiation MOP has many optional scenes and no scene which needs to directly follow the introduction. Julia has a goal to cater, and the reason-for-presence scene is mandatory, but the scene does not need to closely follow the introduction so it does not have enough activation to be selected for execution. The user utterance in the turn-taking MOP must immediately follow a system utterance, so that scene is selected for execution. Since the scene must be executed by the user, JUDIS waits. But as the system waits, the contribution made by the strength of the connection between the scenes decays. Eventually, it decays enough that the decision for the next MOP relies mostly on the goals; "May I help you?" is executed since its goal has more strength than the goal of *polite-conversation* that is associated with turn-taking.

CONCLUSION

JUDIS handles a restricted conversational behavior, but we believe it can be extended to handle much more complicated parts of a dialogue. Some of the power of our conversational controller will be derived from its association with a dynamic memory and with problem solvers. The dynamic memory is useful for finding the appropriate MOP for a given context. The problem solvers can be used to infer general information about the world needed to understand a conversation. Their problem solving capabilities can be used to help plan conversation when active MOPs do not specify enough detail. As we continue our research, the conversation MOPs themselves will need to be expanded as the system handles dialogue that is less constrained. The factors contributing to the activation of a MOP will have to be expanded. We will also have to consider more complicated MOPs, such as those for topic switching.

As we move further into the dialogue, we will need to explore how problem solving affects the use of convention. Here, the intentional component of natural language processing will be even more important and a user's plan will more often need to be inferred. The system will also need to make general MOPs more specific by filling in information from the problem solving context. The problem solving systems and the user will place more demands on JUDIS's flexibility and will require many MOPs to be active at one time.

Although we must continue to develop our system in order to handle more interesting conversational phenomena, our current implementation has given us a solid base to work from. We have used the philosophy of minimizing cognitively implausible reasoning effort in a system that can be integrated with more intensive reasoning when necessary. By using conversational MOPs to represent conversational rules, we have laid the groundwork for a system which can be flexible while retaining the efficiency of following conventions.

ACKNOWLEDGMENTS

Many thanks to Roy Turner and Mark Graves for their helpful comments on earlier versions of this paper.

REFERENCES

Allen, J. & Perrault, C. R. (1980). Analyzing intention in utterances. *Artificial Intelligence 15*, 143–178.

Austin, J. L. (1962). *How to do things with words.* (J. O. Urmson Ed). New York: Oxford University Press.

Carberry, M. S. (1986). *Pragmatic modeling in information system interfaces.* (Tech. Rep. No. 86-07 and Ph.D. thesis). Newark: University of Delaware, Department of Computer and Information Sciences.

Cohen, P. and Perrault, C. R.(1979). Elements of a planning based theory of speech acts. *Cognitive Science 3*(3), 177–212.

Cullingford, R.E. & Kolodner, J.L. (1986). Interactive advice giving. In *Proceedings of the 1986 IEEE International Conference on Systems, Man and Cybernetics.* (pp. 709–714) Atlanta, Georgia.

Douglas, W. (1984). Initial interaction scripts: When knowing is behaving. Paper presented at the Annual Meeting of the International Communication Association, San Francisco.

Grosz, B. J.(1977). The representation and use of focus in a system for understanding dialogs. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence.* (pp 67–76). Los Altos, California: Morgan Kaufmann.

Kellermann, K. & Broetzmann, S., Lim, T., Kitao, K. (in press). The conversation MOP: Scenes in the stream of discourse. To appear in *Discourse Processes.*

Kolodner, J. L. (1984). *Retrieval and organizational strategies in conceptual memory: A computer model.* Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Litman, D. (1986), Understanding plan ellipsis. In *Proceedings of the National Conference on Artificial Intelligence*, (pp. 619–624), Los Altos, California: Morgan Kaufmann.

McKeown, K. R. (1985). *Text generation: Using discourse strategies and focus constraints to generate natural language text.* New York: Cambridge University Press.

Reichman, R. (1985). *Getting computers to talk like you and me: Discourse context, focus, and semantics (An ATN model).* Cambridge, Mass.: The MIT Press.

Schank, R. C. & Abelson, R. P. (1977). *Scripts, plans, goals and understanding: An inquiry into human knowledge structures.* Hillsdale, N.J.: Lawrence Erlbaum Associates.

Schank, R. C. (1982). *Dynamic Memory.* New York: Cambridge University Press.

Searle, J. (1969). *Speech acts: An essay in the philosophy of language* Cambridge, UK: Cambridge University Press.

Sidner, C. (1983). Focusing in the comprehension of definite anaphora. In *Computational models of discourse*, M. Brady and R. Berwick, eds., Cambridge, Mass.: MIT Press.

A Theory of Simplicity

Gilbert Harman, Michael Ranney, Ken Salem, Frank Döring
Jonathan Epstein, Agnieszka Jaworska

Cognitive Science Lab, Princeton University, 221 Nassau St., Princeton, NJ 08542.
609-987-2824, ghh@princeton.edu

The simplicity of a hypothesis for a person cannot be measured by the simplicity of the person's representation of that hypothesis (for example, the number of symbols used), because any hypothesis can be represented with a single symbol. A better measure of simplicity is the ease with which the hypothesis can be used to account for actual and foreseeable data. But it is also important to allow for different ways in which data might be represented. We suggest that the relevant ways of representing data are those ways in which the person is interested, *i.e.*, those representations that most directly help to answer questions the person wants to answer. In particular, we suggest that the simplicity of a hypothesis for a person is determined by the shortness of the connection between that hypothesis and the data that interest the person, as measured by the number of intermediate steps he or she needs to appreciate in order to appreciate the complete connection.

Keywords: simplicity, explanation, inference, hypothesis

Curve Fitting

Suppose that you have discovered that pressure on a certain sort of ceramic material affects the conductivity of that material and you are interested in being able to determine the relation between these two quantities. You obtain the following replicable data pairs: $P=1$, $C=2$; $P=3$, $C=6$; $P=4$, $C=8$; $P=9$, $C=18$. If we let $C=F(P)$, your data can then be expressed as follows: $F(1)=2$; $F(3)=6$; $F(4)=8$; $F(9)=18$. Given these data, you would normally take the most reasonable hypothesis to be (H1) $F(P)=2P$.

Why? Certainly, (H1) is compatible with your data in a way that various other hypotheses are not, for example:

(H2) $F(P)=3P$.

(H3) $F(P)=3-P$.

But there are infinitely many hypotheses that are like (H1) in being compatible with your data. For example,

(H4) $F(P)=2P+(P-1)(P-3)(P-4)(P-9)$.

What is it that makes you prefer (H1) to a hypothesis like (H4)?

The natural answer to this question is that (H1) is simpler and less ad hoc than (H4). But this answer raises two important and related issues. First, what makes one hypothesis simpler than another? Second, why should the simplicity of a hypothesis be taken as any sort of indication of the truth of the hypothesis?

The preparation of this paper was supported in part by research grants to Princeton University from the James S. McDonnell Foundation and the National Science Foundation. We are indebted to discussions with Paul Thagard and (some years ago) with Hugo Margain.

These are difficult issues. We are going to concentrate on the first and try to say what the simplicity of a hypothesis consists in. But we will also say a little about the second issue, why simplicity, so understood, should be used to select among hypotheses that account equally well for the data.

In the example we began with, the first question becomes this: In what way is (H1) simpler than (H4)? One difference between the hypotheses is that (H1) is expressed in 7 symbols while (H4) uses 28 symbols. So we might consider the following proposal: (H1) is simpler than (H4) in that it is shorter, and, more generally, the complexity (or lack of simplicity) of a hypothesis can be measured by the number of symbols used to express that hypothesis (Sober, 1975).

The trouble with this proposal is that hypotheses can be expressed in various ways. For any given hypothesis we can arbitrarily introduce notation that would allow us to express the hypothesis with very few symbols. For example, we have been using the symbols "(H1)" and "(H4)" to stand for these hypotheses, which shows that each can in fact be represented in 4 symbols. We could have used the numerals "1" and "4" instead, in that way representing each hypothesis with one symbol. Clearly, any hypothesis can be expressed using a single symbol in this way. So, if hypotheses differ in simplicity, this difference cannot be measured simply by counting the symbols used to express the hypotheses.

It might be suggested that "(H1)" and "(H4)" are abbreviations of longer expressions and that we must consider how many symbols it takes to express these hypotheses without using abbreviations. But how are we to tell when abbreviations have been used? How are we to determine what symbols can be used as primitive and undefined? Is it permitted to represent the product of 2 and P as "$2P$" or must we include an operator to represent multiplication as in "$2\times P$"? Can we use ordinary mathematical symbols like "×", "+", and "−", or are these symbols to be defined in even more primitive terms?

One way to answer such questions is to postulate a system of mental representation. In this view, the complexity of a hypothesis for you depends on your actual mental representation of the hypothesis. (H1) is less complex than (H4) for you, even though each can be externally represented by a single symbol, if mental representation of (H1) uses fewer mental primitives than your mental representation of (H4). This is a psychological account of simplicity in the sense that the simplicity of a hypothesis is its simplicity for you, given the way in which you represent the hypothesis in Mentalese.

This leaves the difficult problem of trying to determine what form an inner system of mental representation takes. One test might be to consider how "natural" certain representations are. Consider the following two representations:

(H1) $F(P)=2\times P$.

(H4) $F(P)=2\$P$.

Here the operator "×" stands for multiplication and the operator "$" satisfies the following rule:

$$A\$B=A\times B+(B-1)\times(B-3)\times(B-4)\times(B-9).$$

The operator "×" seems quite natural as an operator that is suitably represented by a single symbol, whereas the operator "$" so defined may seem "unnatural"—one that is *not* suitably represented by a single symbol.

But this does not meet the objection already considered, since it does not preclude there being various ways to mentally represent any given hypothesis. Just as you can use a single symbol to stand for any given hypothesis while you are talking to someone else, you ought to be able to do the same thing in the systems of mental representation you use for thought. If you can, the complexity of a hypothesis for you is not a function of the number of mental symbols you use to

express the hypothesis in your inner language of thought.

In other words, it is unclear why you can't introduce a mental symbol for the operation "$" so that you can *mentally* represent (H4) as "$F(P)=2\$P$"? That may not be a natural way of representing (H4) in some sense, since the symbol "$" is unfamiliar, but that fact cannot prevent "$F(P)=2\$P$" from being a possible mental representation of (H4).

One thing that may make this particular representation seem unnatural is that, in order to use it, you need to unpack it. If you are to use this representation of (H4) to calculate the value of "2$7," you need first to translate the representation into the equivalent form "2×7+(7–1)(7–3)(7–4)(7–9)." You can then see that this is equivalent to "14+6×4×3×(–2)," which is equal to "14–6×4×3×2," which is equal to "14–24×6," which is equal to "14–144," which is equal to "–130."

This calculation is more complicated than simply multiplying "2×7." That is what seems to make (H4) a more complicated hypothesis. Notice that the complexity of the calculation has little to do with how (H4) is represented. No matter how the hypothesis is represented, a relatively complicated calculation is needed in order to determine the value $F(P)$ for particular values of P. The complexity of the hypothesis depends not just on how it is represented, but rather on the calculations needed to use the hypothesis.

Let us generalize this idea. Instead of measuring the complexity of a hypothesis by the number of symbols used to represent the hypothesis in either English or Mentalese, we can measure complexity by the amount of processing required to use the hypothesis in order to connect it with the data -- or more generally, the amount of processing that is needed to use the hypothesis in whatever way the hypothesis is to be used. As we have seen, this would count (H1) as simpler than (H4) on the grounds that determining $F(P)$ for various P requires fewer calculations under (H1) than under (H4)

This is still a "psychological" account of simplicity and complexity, but the psychological simplicity or complexity of a hypothesis is taken to depend on the complexity of certain mental processes rather than depending directly on the complexity of particular mental representations. Changing notation to make a hypothesis shorter will not make that hypothesis simpler, unless the new notation allows a psychologically shorter derivation of the data.

Immediately Obvious Steps

The complexity of the connection between a hypothesis and an implication of that hypothesis might be measured by the number of steps needed for a person to recognize that connection. For example, the connection between "2×7" and "14" is relatively immediate once you have mastered the multiplication table for "2". No further calculation is needed to see the connection between "2×7" and "14." There is not an equally immediate connection between "(2×7)+(7–1)×(7–3)×(7–4)×(7–9)" and "–130". You can appreciate this connection only by virtue of noting intermediate steps such as those mentioned above, namely, "14+6×4×3×(–2)," "14–6×4×3×2," "14–24×6," and "14–144."

The relevant steps are "immediately obvious steps." That is, you can recognize that each step follows from a previous step without having to recognize some intermediate steps. We claim that this notion of an "immediately obvious step" is quite important for the theory of inference. It is important not only for understanding simplicity, but for other reasons as well.

Harman (1987) argues that the concepts of logic might be at least partially explicated in terms of immediately obvious steps of implication between propositions involving these concepts. The operation of logical conjunction, "&," has the property that a conjunctive proposition "$A\&B$" implies and is immediately implied by its conjuncts "A" and "B". This is not true of the

more complex but logically equivalent concept "%" explicated as follows: "*A %B*" = "not either not *A* or not *B*." If the meaning of a concept depends in part on the immediately obvious implications of propositions involving the concept, logically equivalent concepts can differ in meaning.

It is worth noting that steps that are not immediately obvious at one time can become immediately obvious at some later time. For example, at one time it may not have been obvious to you that 4×7 is 28. You may have had to note such intermediate steps as these: 4×7; 7+7+7+7; 14+7+7; 21+7; 28. At a later time, you can immediately see the equivalence. Your recognition of it does not depend on the conscious recognition of such intermediate steps. In our view, when you came to acquire this sort of mathematical skill, hypotheses involving multiplication became psychologically simpler for you. (And your concept of multiplication underwent a change.)[1]

Goodman's Grue Bleen Problem

We suggest further that the complexity of a hypothesis is relative to what you are interested in, that is, it depends on the complexity of the connections between the hypothesis and things in which you have an interest. In order to illustrate this suggestion, let us look at one aspect of Goodman's famous grue bleen puzzle (Goodman 1965), which is a variant of the curve fitting problem.

Suppose that you have examined a variety of emeralds and have determined in each case that the color of the emerald was green at the time of observation. Suppose that it is now January 1, 1990 and consider the following two hypotheses:

(H5) All emeralds are (always) green.

(H6) All emeralds *either* (a1) are first observed before the year 2000 and (a2) are (always) green *or* (b1) are not first observed before the year 2000 and (b2) are (always) blue.

These hypotheses agree about all emeralds first observed before the year 2000 but disagree about emeralds that do not get observed by then. (H5) implies that such emeralds are green. (H6) implies that such emeralds are blue.

Although both hypotheses would account for your evidence, you prefer (H5) over (H6) on grounds of simplicity. The measure of simplicity we have proposed seems to account for this, since the connection between any item of evidence and (H5) would seem to be immediate whereas the connection between an item of evidence and (H6) would seem to be mediated by several steps:

> All emeralds are either observed before the year 2000 and are (always) green or are not observed before the year 2000 and are (always) blue. So, this emerald is either observed before the year 2000 and is (always) green or is not observed before the year 2000 and is (always) blue. This emerald is observed before the year 2000 So, it is not

[1] In supposing that the equivalence between 4×7 and 28 is immediately obvious, representing a "single step," we do not deny that there may be a level of analysis at which the recognition of this equivalence involves several steps. For example, you could look up this product in a table in your memory. You could first locate the table, then locate the relevant column and row. It is possible that, at this deeper level of analysis, the recognition of the equivalence between 4×7 and 28 requires more steps and so more time than the recognition of the equivalence between 2×10 and 20, even though both of these equivalences are immediately obvious at a coarser level of analysis. [This issue is discussed in, e.g., Groen & Resnick (1977) and Ashcraft & Stazyk (1981).] We do not at this time know how to give a precise characterization of the relevant deeper level of analysis. All that concerns us at this stage is the difference between your situation before and after you have mastered the times tables. [Another complication is that a step of inference might be immediate even though the conclusion of that step is itself a complex argument-structure involving several steps of implication or explanation. See Harman et al. (1987).]

the case that this emerald is not observed before the year 2000 So, it is not the case that this emerald is not observed before the year 2000 and is (always) blue. So, this emerald is observed before the year 2000 and is (always) green. So, this emerald is green.

But more needs to be said about this case. Following Goodman, you can define a predicate "grue" as follows:

x is grue at *t* if and only if either (1) *x* is first observed before the year 2000 and is green at *t* or (2) *x* is not first observed before the year 2000 and is blue at *t*.[2]

Then you can represent this last hypothesis more briefly as follows:

(H6) All emeralds are (always) grue.

Since all emeralds observed so far are observed before the year 2000, all the observed emeralds are grue (at least when observed). And, while it is true that the hypothesis that all emeralds are green offers a simpler account of the fact that observed emeralds have so far been green, it is also true that the hypothesis that all emeralds are grue offers a simpler account of the fact that observed emeralds have so far been grue. Your situation with respect to the two hypotheses may therefore seem perfectly symmetrical, and it becomes unclear how you can take (H5) to be any simpler than (H6).

Here there is a temptation to return to counting symbols to measure complexity. Although you can use the term "grue" in order to express (H6) as compactly as (H5), it might be argued that this compact representation is an abbreviation of your original, much more complicated, representation of the hypothesis. In this view, "green" expresses a simple concept, whereas "grue" expresses a disjunctive concept.

But this attempt at a solution leads to the difficulties mentioned earlier. There is no objective way to determine when a representation should be counted as an abbreviation. Furthermore, Goodman points out that "green" and "blue" can be seen to express disjunctive concepts via the following definitions (where "bleen" is understood complementarily to the way in which "grue" is understood).

x is green at *t* if and only if either (1) *x* is first observed before the year 2000 and is grue at *t* or (2) *x* is not first observed before the year 2000 and is bleen at *t*.

x is blue at *t* if and only if either (1) *x* is first observed before the year 2000 and is bleen at *t* or (2) *x* is not first observed before the year 2000 and is grue at *t*.

It might be suggested that "green" is a term that can be applied purely on the basis of observation whereas "grue" is not, and that this has something to do with why we prefer (H5) to (H6). However, it is unclear how to make a general distinction between observational terms and other terms. (Is "emerald" an observational term?) Furthermore, it is an accident of this particular example that it involves a term that might count as observational. Many hypotheses in which we are interested do not make use of observational terms in this way and the same problem arises for them as for (H5). That is, we are not primarily concerned with hypotheses of the form, "All A's are B's," where "A" and "B" are observational terms. A distinction between what is directly observable and what is not cannot yield a general solution of Goodman's puzzle.

As we have already said, our proposal is that interests have a bearing on inferences. What conclusion you should reach depends in part on what question you are interested in answering.

[2] Notice that there are two time references, the time *t* at which we are considering what color *x* is and the usually quite different time at which *x* is first observed.

Normally, you are interested in what is blue or green, not in what is bleen or grue. (H5) is preferable to (H6) because (H5) gives a simpler route to the sorts of things you are interested in than (H6) does. Although (H6) provides a simpler route to conclusions about what things are grue or bleen, you are normally not directly interested in learning such things.

So, we suggest that the simplicity of a hypothesis for you is determined by the simplicity of the connection between that hypothesis and the data in which you are interested, as measured by the number of intermediate steps you need to consider in order to see the connection. Since you will sometimes accept a hypothesis just because it is the simplest of a group of hypotheses that equivalently account for the data, this suggestion implies that your interests can influence what conclusion you come to accept. A natural objection is that this must be a case of irrational wishful thinking. What can we say about that objection?

Wishful Thinking and the Relevance of Interests to Inference

Here it is important to distinguish reasoning that is aimed at what to believe, which (following Aristotle) we can call "theoretical reasoning," from reasoning that is aimed at what to plan to do, which we can call "practical reasoning." Clearly, your interests can legitimately help to determine what practical conclusions you should reach about what to do, so that is an obvious way in which your interest are relevant to your reasoning. But theoretical reasoning is not practical reasoning and we are now concerned with how your interests might affect what theoretical conclusions you are justified in reaching.

Consider a related way in which your interests can be relevant to your theoretical reasoning. Your interests help to determine what questions you have reasons to answer. In that way, your interests can legitimately affect which conclusions you will draw. At any moment, a vast number of conclusions follow trivially from your beliefs. But you are not equally justified in drawing each of those conclusions, since at best you will be interested in the truth of only a small number of them. Reasoning is subject to a principle of clutter avoidance. You should not clutter your mind with the trivial consequences of your beliefs, at least if there are certain questions you might be resolving in which you have an interest. This is not a general warrant for wishful thinking. The fact that you want a certain result to be true is not a reason to believe that it *is* true. Your interests can give you a reason to try to answer a particular question but they are irrelevant to what the answer is (except in special cases, for example, in which the question concerns your interests).

This is relevant to our proposal for resolving the grue-bleen problem. Our resolution appeals to your interests in order to determine what questions you want to answer, not what the answers are. We suggest that it is legitimate for you to accept the simplest account of the data in which you are interested, where simplicity is measured by the number of steps needed to get from hypothesis to data. You tend to be interested in whether certain things are blue or green, not in whether they are grue or bleen. You are normally interested in why observed emeralds are green in a way in which you are not so interested in why observed emeralds are grue. Even philosophers who are interested in why certain emeralds are grue are interested only because of an ultimate interest in whether unobserved emeralds are green. It is their ultimate interest in answering questions about green and blue that leads philosophers and others to accept (H5) rather than (H6). You do not accept (H5) because you prefer the answers that (H5) gives to other answers. Whether you want emeralds to be green or blue is irrelevant.

But what about the appeal to *simplicity* in believing (H5) rather than (H6)? Is that a case of wishful thinking? It is true that simpler hypotheses have pragmatic advantages over more complex hypotheses in that they are easier to use in accounting for data and in making predictions. So, you have a practical reason deriving from your interests to prefer believing simpler

hypotheses over believing more complex hypotheses. We (the authors of this paper) are divided as to whether believing a simpler hypothesis for this sort of practical reason is to engage in wishful thinking. A standard case of wishful thinking involves believing one hypothesis rather than another *because you want the first hypothesis rather than the second to be true*. Now, to prefer the simpler hypothesis because it is easier to use need not involve wanting that hypothesis to be true. A preference for believing *X* over believing *Y* is not the same as a preference for *X*'s being true over *Y*'s being true. Although this is not a standard case of wishful thinking, it may be just as bad.

So, we are still left with the second of the two questions about simplicity with which we began. The first question was "What makes one hypothesis simpler than another?" We have made a proposal about that. The second question is "Why, given that way of measuring simplicity, should you take the simplicity of a hypothesis to be any sort of indication of the truth of the hypothesis?" That is a deep question and we do not have the space for a full-scale discussion. All we can say is that we are unclear as to whether there is an independent source of information about what is likely to be true over and above the principles of reasoning we actually follow. Since people use simplicity to decide among hypotheses that are otherwise equally satisfactory, when we reflect on particular cases of this sort, the simpler hypothesis is likely to seem the most reasonable conclusion (unless we are temporarily skeptical).

Given a set of hypotheses that all account for the data, we do in fact take the simplicity of a hypothesis as making that hypothesis more likely than less simple alternatives. That is what we do, and we do not seem to have any reason to stop. Perhaps we are justified in continuing to use simplicity in this way—in the absence of a serious difficulty with our current practice and the absence of any reasonable alternative.

Bibliography

Ashcraft, Mark H., & Stazyk, Edmund H. (1981). "Mental addition: a test of three verification models." *Memory & Cognition, 9*, 185-196.

Goodman, Nelson (1965). *Fact, Fiction, and Forecast*, 2nd edition. Indianapolis: Bobbs, Merrill.

Groen, Guy, and Resnick, Lauren B. (1977). "Can preschool children invent addition algorithms?" *Journal of Educational Psychology, 69*, 645-652.

Harman, Gilbert (1986). *Change in View: Principles of Reasoning*. Cambridge, Massachusetts; M.I.T./Bradford Books.

Harman, Gilbert (1987). "(Nonsolipsistic) conceptual role semantics." In Ernest LePore (ed.), *New Directions in Semantics*. London: Academic Press, 55-81.

Harman, Gilbert, Bienkowski, Marie A., Salem, Ken, & Pratt, Ian (1987). "Measuring change and coherence in evaluating potential change in view." *Ninth Annual Conference of the Cognitive Science Society*. Hillsdale, NJ.: Erlbaum, 203-209.

Sober, Elliot (1975). *Simplicity*. London: Oxford University Press.

Basic Levels in Hierarchically Structured Categories

James E. Corter
Columbia

Mark A. Gluck
Stanford

Gordon H. Bower
Stanford

In a hierarchical set of categories, one level of the hierarchy is said to be "basic" if categories at that level seem to be used more easily (e.g. named faster) than categories at other levels. An experiment was conducted to determine whether the feature structure of a hierarchy of artificial categories (fictitious diseases) can induce "basic level" effects. The feature structure of this hierarchy was patterned after an experiment by Hoffman and Ziessler (1983), in which the middle level of the hierarchy was found to be "basic". In a learning task, subjects learned to diagnosis patient descriptions (consisting of a list of symptoms) in terms of the fictitious diseases. In a subsequent verification task, they were shown a patient description paired with a diagnosis, and asked to judge the correctness of the diagnosis. Mean reaction times were smallest for categories at the middle level of the hierarchy, showing that this level was indeed basic. In addition, analysis of data from the learning task showed that subjects made the highest proporation of correct diagnoses for diseases at the middle level, thus confirming the advantage of the middle level. These results provide additional evidence for basic level phenomena in the retrieval of categorical information from memory for hierarchically structured categories; they extend previous findings by showing basic level effects with categories described in terms of conceptual (verbal) rather than purely visual features.

INTRODUCTION

Certain categories seem to be learned and used more easily than others. One line of research that has shown this clearly is the work on "basic level" categories (Mervis & Rosch, 1981). Within a hierarchy of categories at various levels of generality (e.g. *sparrow-bird-animal*), one particular level of category (*bird* in this example) is usually found to be used most readily, and is referred to as the *basic level*. In laboratory studies the criterion variable most often used to determine which level of a hierarchy of categories is basic is the mean reaction time to name categories at that level, or to verify category-name pairings (Rosch, Mervis, Gray, Johnson, & Boyes-Braem, 1976; Jolicoeur, Gluck, & Kosslyn, 1984).

Rosch and colleagues (e.g., Rosch & Mervis, 1975) originally suggested that basic level categories are special because they capture significant regularities or patterns in the features associated with these categories. For example, basic level categories appear to be the most general (i.e., abstract) categories for which members have many common properties. We refer to this type of explanation of the basic level advantage as a *structural* theory, since it implies that certain categories are "basic" because of their structural properties, namely, the statistical associations between features and categories.

An alternative to such structural theories of the basic level phenomenon is the notion that *type* of feature is important in determining the basic level. For example, Tversky and Hemenway (1984) reported that when people are asked to give lists of features for various categories, there is a difference in the *type*

Correspondence should be addressed to: Dr. James E. Corter, Box 41, Teachers College, Columbia University, New York, NY 10027. This research was supported by NSF Grant BNS-8618049.

of features they give for superordinate and basic level categories. For superordinate categories, associates and functions are commonly listed. At and below the basic level, the most commonly mentioned features of concepts are *parts*, such as *wings* for birds and *engine* for cars.

Thus the existence of basic levels may be due to qualitative differences in the features associated with categories rather than to quantitative differences. Perhaps more abstract, superordinate categories (e.g. *vehicles*, *weapons*) aggregate objects by similar *functions* which are abstract and nonperceptual, whereas basic level categories rely more heavily on common perceptual properties. According to this *feature-type* hypothesis, the basic level findings simply show that perceptual features are more easily used than abstract features such as function.

An important experiment by Murphy and Smith (1982) demonstrated that basic-level effects could be obtained by manipulating the feature composition of a set of artificial stimuli. In this experiment subjects were taught a hierarchy of artificial categories (fictitious tools) that was designed so that the middle level should be basic. In later naming and verification tasks, their subjects responded quickest to categories at the middle level, thus confirming that this level was psychologically "basic". However, Murphy and Smith did not design their experiment to distinguish between the feature-structure and feature-type explanations. While their subordinate and basic level categories were distinguished by perceptual features, their superordinate categories were defined only in terms of function (tools used for pounding vs. tools used for cutting).

This confounding of level with type of feature was avoided in experiments by Hoffman and Ziessler (1983). Different groups of subjects were taught one of three category hierarchies. Exemplars of the categories were schematic line drawings of rocket ships. In one hierarchy, the feature-category associations were such that the top level was expected to be basic, in the second hierarchy so that the middle level would be basic, and in the third so that the bottom level would be basic. These predictions were confirmed by subjects' reaction times to name the categories and to verify category-name pairings. In each hierarchy the categories at the basic level were named fastest, regardless of whether the basic level corresponded to the top, middle, or bottom level of the hierarchy.

The results of these last two experiments demonstrate that feature structure and/or feature type are sufficient to cause a basic-level effect. However, because feature-structure and feature-type were confounded by Murphy and Smith, only the Hoffman and Ziessler experiments demonstrate that feature structure alone is sufficient to cause a basic-level effect. It is important that this critical experiment be replicated and extended. In particular, since pictorial stimuli were used in both previous studies, the question remains of whether it is only visual features that can induce basic-level effects. Therefore, the present experiment was designed to see if basic level effects could be obtained with non-pictorial stimuli, specifically, categories defined in terms of verbal features. If so, this would indicate that the basic level effects are not purely a perceptual phenomenon, but rather arise from very general cognitive mechanisms. Such a finding would thus constitute strong evidence that a purely structural explanation of basic level phenomena is tenable.

EXPERIMENT 1

A set of materials was designed to mimic the feature structure of Hoffman and Ziessler's Hierarchy 2. However, our categories were defined in terms of verbal or conceptual features rather than perceptual ones. Specifically, the categories were diseases and instances of the categories were individual patients with a set of symptoms characteristic of the disease. In a training task subjects learned to diagnose individual patients, each described as a set of symptoms, in terms of which disease the patient had.

Following the training task a verification test was given in which subjects were presented with a variety of patient descriptions paired with a diagnosis, and were asked to indicate whether that diagnosis was correct. The measure of primary interest was the average reaction time to verify diagnoses for categories at various levels of the hierarchy.

Materials

The concepts to be learned were fictitious diseases. Instances of the diseases were individual patient descriptions, each consisting of a list of three symptoms. An example patient description is: "blotchy rash, swollen gums, red eyes". One symptom always pertained to gums, one to eyes, and one to rash. Each of these symptom "dimensions" was substitutive in nature, having four possible values. An individual patient description contained one value of each of the three symptom dimensions. The four possible values of the rash dimension were: *blotchy, spotted, itchy, scaly*. For gums they were *swollen, discolored, bleeding, sore*, and for eyes they were *puffy, sunken, red, burning*.

The set of diseases to be learned were defined at several levels of generality. Specifically, the concepts formed a hierarchy with eight bottom-level, four middle-level, and two top-level categories. The hierarchy of disease names is shown in Table 1. The feature structure (i.e. the symptom-disease associations) of these categories was adopted from Hoffman and Ziessler's (1983) Hierarchy 2, which was designed so that the middle level was expected to be psychologically basic. The stimuli in Table 1 were made up to have this same structure, with symptom dimensions replacing the perceptual feature dimensions of the Hoffman & Ziessler stimuli.

TABLE 1: The Disease Hierarchy and its Feature Structure

Categories			Feature Dimensions		
Top	Middle	Bottom	Gums	Eyes	Rash
		-- jirenza	1	1	1
	-- burlosis --<				
|		-- malenza	1	2	1
queritism ---|					
|		-- gilenza	2	3	2
	-- cretosis --<				
		-- surenza	2	4	2
		-- habenza	3	1	2
	-- midosis --<				
|		-- kelenza	3	2	2
philitism ---|					
|		-- tumenza	4	3	1
	-- nitosis --<				
		-- valenza	4	4	1

For each subject the correspondence between features and structure was randomly assigned. Also, middle-level and bottom-level category names were randomly assigned to the feature structure shown.

Method

Subjects were students and other members of the Columbia University community who were paid $6 per hour for their participation. In all, 27 subjects were tested. Subjects were informed that each patient description would consist of one rash symptom, one eyes symptom, and one gums symptom. It was explained that just as real diseases could be diagnosed at more or less general levels (e.g. jaundice, hepatitis, hepatitis A), so the present diseases would be identified sometimes in more general and sometimes in more specific terms. However, the actual hierarchy of diseases was never shown to the subject.

Stimuli were presented, and responses collected, on an IBM AT microcomputer. The training phase involved blocks of two different tasks: study and testing. In a given block diagnoses were to be made only at a single level of the hierarchy. In a study block, the subject saw a succession of patient descriptions, each paired with the correct diagnosis. Immediately after this a testing block was given at the same level. Here the subject saw patient descriptions and had to identify them with names of diseases at that level. A response was entered by typing the first letter of the disease name. Corrective feedback was given following the subject's response. Both the study and testing were self-paced, although subjects going extremely slow during the first few study blocks were urged to speed up. Blocks were of different lengths, depending on the level of categorization, in order to equalize the frequency of usage of each category (disease) name. Top-level blocks consisted of eight trials (one presentation of all eight patterns of symptoms), middle-level blocks had 16 trials (two cycles through the eight patterns), and bottom-level blocks had 32 (four cycles). Thus, each category name was experienced four times per block.

Overall, the structure of the training phase was as follows. Subjects were given a study block followed by a test block at one level, then a pair of blocks at another level (i.e., a study followed by a test block), then a pair at the third level. Then three more pairs of blocks were given, using a different ordering of levels. At this point the subject had seen six study blocks and six test blocks. Six more test blocks followed. If by this time the subject was performing at criterion (90% correct diagnoses) on all three levels, the training was terminated. Otherwise the subject repeated sets of three test blocks (one at each level) until this overall criterion was attained. Three different orderings of blocks were counterbalanced across subjects. One order presented the bottom level first, one the middle level, and one the top level. The orders were: M-T-B-T-B-M-B-M-T-..., T-B-M-B-M-T-M-T-B-..., B-M-T-M-T-B-T-B-M-...

Once subjects had learned the category hierarchy to criterion, their use of these categories was tested in a verification task. In the verification task subjects were presented with a patient description, paired with a correct or an incorrect diagnosis. The subject responded "yes" or "no" by means of specially labelled keys on the keyboard. Simple feedback on the validity of the subject's response ("correct" or "incorrect") was given. Half the trials were true ("yes") trials, and half were false ("no") trials. The reaction time and correctness of each response was recorded. There were 12 blocks of 24 trials each. As in the training phase, in a block of trials diagnoses (both correct and incorrect) were made only at a single level. The same three orderings of levels described above were used in this task.

Results

Six subjects (out of 27) did not learn the categories to criterion within the allotted time (2.5 hours) and therefore were not given the verification task. Thus complete data was available for 21 subjects.

The measure of primary interest for identifying the basic level is the mean reaction time to verify diagnoses at each of the three levels in the verification task. Responses should be quickest for categories (diseases) at the basic level. In the verification task there were four blocks at each level. The first of these was not analyzed. Based on an examination of the distribution of reaction times for individual subjects, responses with latencies greater than 15 seconds were considered to be outliers and were excluded from further analysis. Less than 1% of the responses were eliminated by this rule. The mean reaction times for each level (averaged over the three analyzed blocks) are shown in Table 2.

TABLE 2: Mean and S.D. of Verification Reaction Times by Level of Categorization

	TOP	MIDDLE	BOTTOM	OVERALL
MEAN	3.115	2.567	3.045	2.909
S.D.	1.313	0.679	0.858	1.001

The hypothesis that the middle level would be verified fastest was tested with a special contrast in a repeated-measures ANOVA. The hypothesis was confirmed, $F(1,20) = 14.20$, $p < .01$. A comparison of the top and bottom levels was not significant, $F(1,20) = 0.05$. A majority of subjects (11 out of 21) showed the expected pattern (that is, verified the middle-level categories fastest). Five subjects verified the top-level categories fastest, and five the bottom level categories.

The learning task can also provide useful data on the relative "goodness" of categories at the three levels of the hierarchy, in the form of the proportion of correct category identifications at each level of the hierarchy. If the middle level is indeed basic for this structure, then a higher proportion of middle-level categories should be correctly identified, compared with top-level or bottom-level categories. This hypothesis was confirmed by the data. Across all blocks, the mean proportion correct was .824 for the top level, .924 for the middle level, and .878 for the bottom level. The advantage of the middle level over the top and bottom levels was tested by a special contrast in a repeated measures ANOVA, and was significant at the .01 level, $F(1,20) = 8.96$. Top and bottom levels did not differ significantly, $F(1,20) = 1.51$. The ordering of levels by proportion correct did not change when the proportions were corrected for guessing by the formula $p' = p - \frac{(1-p)}{(k-1)}$, where p' is the corrected proportion, p is the uncorrected proportion of correct identifications, and k is the number of alternative categories at that level. The corrected proportions for the top, middle, and bottom levels were .648, .899, and .861.

The mean proportion correct for a level is calculated over all blocks of a subject's data. Thus, it is an average of a wide range of values, which generally increase toward unity. It is natural to ask if the relative advantage of the middle level varies as a function of block, perhaps decreasing as all levels are learned to criterion. Figure 1 presents the mean proportion correct for each level, calculated by block for the first four blocks (which were the only blocks experienced by every subject).

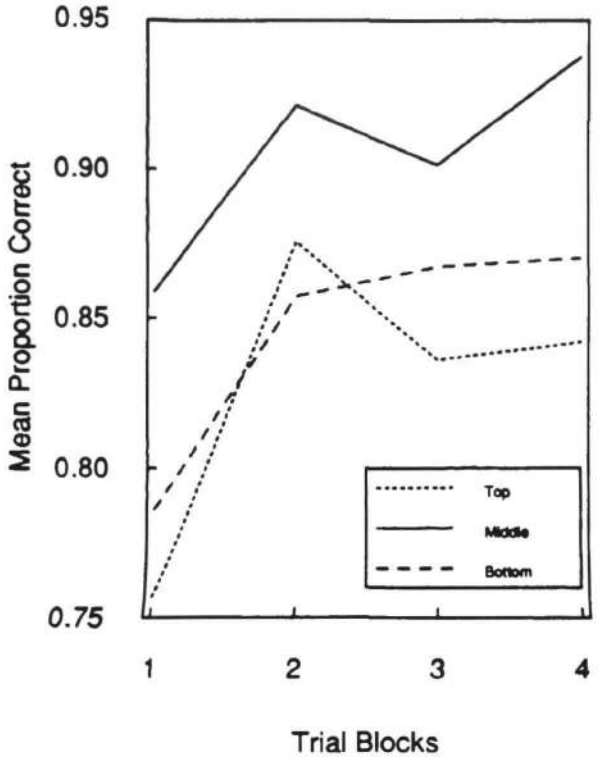


Figure 1. Mean proportion correct for levels by block (first four blocks) from Experiment 1.

It can be seen from Figure 1 that the middle level exhibits an advantage over the top and bottom levels in all blocks. A drop in accuracy for the top and middle levels from Block 2 to Block 3 is evident, no doubt due to the fact that each of the first two test blocks was immediately preceded by a study block.

Analyses were also done on the learning data from the six subjects who did not learn the diseases to criterion in the allotted time. Average proportion correct for these subjects showed the same pattern (highest accuracy on the middle-level categories) as the subjects who successfully learned the categories to criterion.

SUMMARY

In the present experiment naive subjects were taught a hierarchical set of artificial categories, then tested in a verification task. Mean reaction times in the verification task indicated that the middle level of the stimulus hierachy was basic, since diagnoses were made most quickly at that level. In addition, fewest errors were made on the middle level during the training task, indicating that the middle level was easiest to learn.

The results of the present experiment demonstrate that the feature *structure* alone of a hierarchical set of categories is sufficient to induce a basic level effect, thus replicating the experiments of Hoffman and Ziessler (1983). These results demonstrate that explanations of basic levels based on feature *type* are not necessary. In addition, the present experiment extends previous findings by demonstrating basic level effects with categories described in terms of verbal rather than schematic visual features. Accordingly, the present results provide evidence that the basic level effects are not purely a perceptual phenomenon, but rather reflect more general characteristics of human learning and memory. This provides an additional empirical constraint on models of human category learning. We are currently

working on an extension of our adaptive network model (Gluck & Bower, 1988) to account for basic level effects, but these results are at too early a stage to report here.

REFERENCES

Gluck, M. A., & Bower, G. H. (1988). Evaluating an adaptive network model of human learning. *Journal of Memory and Language, 27*, 166-195.

Hoffmann, J., & Ziessler, C. (1983). Objectidentifikation in künstlichen Begriffshierarchien. *Zeitschrift für Psychologie, 194*, 135-167.

Jolicoeur, P., Gluck, M., & Kosslyn, S. (1984). Pictures and names: Making the connection. *Cognitive Psychology, 16*, 243-275.

Mervis, C., & Rosch, E. (1981). Categorization of natural objects. *Annual Review of Psychology, 32*, 89-115.

Murphy, G. L., & Smith, E. E. (1982). Basic level superiority in picture categorization. *Journal of Verbal Learning and Verbal Behavior, 21*, 1-20.

Rosch, E., & Mervis, C. (1975). Family resemblances: Studies in the internal structure of categories. *Cognitive Psychology, 7*, 573-603.

Rosch, E., Mervis, C., Gray, W., Johnson, D., & Boyes-Braem, P. (1976). Basic objects in natural categories. *Cognitive Psychology, 8*, 382-439.

Tversky, B., & Hemenway, K. (1984). Objects, parts, and categories. *Journal of Experimental Psychology: General, 113*, 169-193.

FLEXIBLE NATURAL LANGUAGE PROCESSING AND ROSCHIAN CATEGORY THEORY

Sandra L. Peters, Stuart C. Shapiro, and William J. Rapaport

Department of Computer Science
State University of New York at Buffalo

1. INTRODUCTION.

Artificial intelligence systems typically hand-craft large amounts of knowledge in complex, static, high-level knowledge structures. These systems generally work well in very limited domains, but are simply too rigid to support natural language understanding in general. This is due in part to the fact that artificial intelligence, natural language processing (AI/NLP) systems have not taken seriously the principles of categorization first set forth in the seminal work of Rosch [1976, 1978, 1981], and later extended by researchers including Barsalou [1987], Murphy & Medin [1985], Lakoff [1987], and Neisser [1987]. Thus, AI/NLP systems have very shallow representations of generic concepts and categories, employing either static, simple, featural models of concepts based on necessary and sufficient criteria in uniform taxonomies where no level is distinguished [e.g., KLONE, Brachman 1983], or passive data structures with slots and explicit default values, such as frames, schemata, and scripts [e.g., KRL, Bobrow & Winograd 1977ab; NETL, Fahlman 1979]. We have previously shown that systems that use the former representations are unable to model human category systems [Peters & Shapiro 1987ab]. In this paper, we discuss the inadequacy of systems based on the latter types of representations, arguing that frames, schemata, and scripts lack the flexibility, generality, and adaptability necessary for representing generic concepts in memory. We present alternative "active" representations, in which frames or schemata do not reside in semantic memory, but rather are constructed as needed from a less organized semantic memory. Their construction can, therefore, be influenced by the current task and context.

In addition, our processing and representations are based on a Roschian model of categories, i.e., on (1) a recognition of the unique nature of basic level categories within natural category systems, and (2) prototype theory. We will discuss some of the current research that supports our representations and processing, and show that our system's performance is enhanced by taking these principles of categorization seriously. Our implementation uses the SNePS knowledge representation and reasoning system, including a generalized ATN parser-generator [Shapiro 1979, 1982; Shapiro & Rapaport 1987]. We present a detailed example that shows the use of our representations and processing strategies in the task of discourse comprehension. In particular, our example will concern implicit focusing in natural language comprehension, i.e., the implicit activation of thematic associates and salient attributes of concepts during discourse comprehension.

1.1. Roschian Model: Basic Level Primacy. Psychology, linguistics, and anthropology have produced a variety of measures of perception, behavior, and communication showing a convergence of cognitive tasks at the basic level. Not all levels of a taxonomy are equally used and useful: for taxonomies of common objects and organisms, the basic level, the level of *table* and *bird*, is the most informative and useful [Rosch et al. 1976; Berlin 1978; Tversky & Hemenway 1984]. Our knowledge is organized at this level (i.e., most attributes of category members are stored at the basic level) and visual imagery is particularly strong for basic level concepts. The informativeness of the basic level originates from the amount of knowledge stored at this level and the rich perceptual component of basic level categories. Basic level concepts trigger many reflex inferences; i.e., they routinely activate many satellite concepts. Where informativeness is the greatest, so too is inferential power.

We have previously presented a representation for natural category systems that uses distinct representations for basic and non-basic level concepts, and discussed empirical evidence supporting basic level primacy and the need for AI/NLP systems to recognize the uniqueness and importance of the basic level [Peters & Shapiro 1987ab]. The work described in this paper builds on, and presupposes that earlier work.

1.2. Roschian Model: Prototype Theory. Although categories have been viewed traditionally as concepts established by necessary and sufficient criteria, and many AI systems continue to model natural concepts in this way [e.g., Brachman 1983], recent categorization research does not support this view [Rosch 1976, 1978; Mervis & Rosch 1981; Murphy & Medin 1985; Barsalou 1987; Lakoff 1987; Neisser 1987]. Rosch has suggested that

another way to achieve the separateness and clarity of categories is by conceiving of each category in terms of its clear cases, i.e., prototypes. Categories possess graded structure: the members of a category vary in how typical they are of their category [Rosch, et al. 1976; Barsalou 1985; Lakoff 1987].

1.3. Extending the Roschian Model. Although Rosch [1978] cautioned that the existence of prototype effects merely indicates that prototypes must have some place in theories of representation of categories, many have misinterpreted her work, construing prototypes as a complete theory of representation for categories and generic concepts. In particular, many AI researchers [e.g., Minsky 1975; Bobrow & Winograd 1977b; Fahlman 1979; Schank 1977] have considered prototypes to constitute a representation for generic concepts, and have employed passive data structures with slots and explicit default values to directly represent these concepts in memory. Thus, they have used frames, scripts, and schemata as a basis for models of cognitive processing, i.e., using them to represent generic objects, situations, events, sequences of events, actions, and sequences of actions. Many cognitive scientists have recently pointed out that passive representations such as frames are too rigid, lacking the flexibility, generality, and richness required in cognitive processing [e.g., Barsalou 1987; Kintsch 1987; Lakoff 1987; Neisser 1987]. Both Schank [1985] and Minsky [1986] have also recently proposed the need for more flexible, dynamic representations. Empirical evidence supporting the need for active representations of generic concepts and categories will be discussed in the following sections.

2. EMPIRICAL EVIDENCE.

2.1. The Instability of Graded Structure of Categories. Barsalou has demonstrated that the graded structure of a category is unstable, varying greatly across contexts. He concludes that "different concepts temporarily represent the same category in working memory on different occasions"[1987, p. 101]. A category's graded structure shifts as a function of (1) the linguistic context and (2) the point of view from which it is perceived [Barsalou 1987]. For example, when *animal* is processed in the context of farm, *cows, horses, goats,* and *pigs* are more typical than *bears, lions, elephants,* and *giraffes.* This situation is reversed when *animal* is processed in the context of zoo.

2.2. The Instability of Generic Concepts. Barsalou [1987] proposes that there are no invariant concepts in long-term memory; rather, long-term memory contains large amounts of highly interrelated knowledge that is used to construct concepts in working memory. Generic concepts are not retrieved intact from long-term memory when needed: they are constructed from long-term memory for a particular task in a particular context. In Barsalou's view, generic concepts have both context-independent and context-dependent information associated them. Barsalou [1982] has presented evidence that shows that context-independent properties are automatically activated by presented concepts, while context-dependent properties are activated only in particular situations or contexts. Thus, the representations for a generic concept in working memory may vary widely across contexts.

2.3. The Inflexibility of Scripts and Schemas. Many researchers have pointed out that computationally fixed mental structures such as scripts are too inflexible to serve the purposes for which they were originally designed [e.g., Schank 1985, van Dijk & Kintsch 1983]. Barsalou [1987] has found that the variables active for a schema vary across contexts in which a schema is used. Similarly, the different tracks that people construct for scripts appear to vary widely across contexts. This may be because these structures exist only as temporary constructs in working memory rather than as invariant structures in long-term memory. Research by Kintsch and Mannes [1987] further supports the idea that scripts do not exist as invariant structures in long-term memory. They claim that knowledge is not pre-organized in terms of scripts and schemata, but that such structures are generated from an unorganized associative net in response to a specific task demand in a specific context. Only in this way, they believe, can the flexibility and context sensitivity that characterize human script use be achieved.

2.4. Summary. An active, flexible representation of generic concepts seems necessary to support the empirical evidence that (1) graded categories, generic concepts, and scripts are not invariant structures; i.e., they vary with context. Our representations and processing are based on an active model of generic concepts.

3. THE STRUCTURE OF BASIC LEVEL CONCEPTS.

3.1. Parts and Basic Level Categories: Physical Object Categories. Tversky and Hemenway [1984] have demonstrated that the basic level differs qualitatively from other levels in taxonomies of objects and living things: part terms predominate in subjects' listings of attributes characterizing objects at this level. Parts are rarely used to characterize superordinate level categories, and members of different subordinate categories share parts and differ on other attributes. Thus, it appears that our knowledge at this level is organized around part-whole divisions. Berlin has also suggested that the perception of overall part-whole configuration is the fundamental determinant of the basic level [1978].

3.2. Perceptual Grounding of Basic Level Concepts. Basic level concepts consist of more than mere attribute lists of parts that we can talk about, however: they are perceptually grounded concepts. Thus, it seems clear that a geometric (3D) model of a basic level object's shape is also encoded in long-term visual memory so that it may be recognized on subsequent occasions [Marr 1982]. Physical objects have their own intrinsic axes used in the 3D model to determine the internal layout of parts: e.g., an up/down axis, a front/back axis, the center/periphery, the interior/exterior. This geometric model enriches these concepts. Lakoff characterizes this type of geometric information as image schematic structure. In discourse, we frequently find references to this image schematic structure; e.g., to the *front* of a house (front/back image schema), the *interior* or *exterior* of a car (interior/exterior schema), and the *top* or *bottom* of many objects (up/down schema).

Additional percepts, e.g., sounds, colors, odors, also characterize objects at the basic level. Thus, if a *dog* is mentioned, its *bark* is implicitly evoked, just as part-whole structures such as *legs, ears,* a *nose,* and a *tail* are evoked.

We depend on both perceptual and functional qualities to differentiate basic level objects into classes. Perceptual properties that have high diagnosticity may be salient, since they are useful for distinguishing instances of a concept from instances of other concepts. Functional properties relevant to how people typically interact with instances of a concept are likely to be highly salient as well.

3.3. The Representation of Basic Level Concepts in SNePS. We use default generalizations [Peters & Shapiro, 1987a] to represent facts about the typical exemplars or members of a category. Thus, a basic level concept in our semantic network is, in part, a collection of default generalizations about typical exemplars: generalizations about their (1) part-whole structure and image schematic structure that is derived from a geometric model in visual memory, (2) other perceptual structure, and (3) functional attributes. These attributes and structures are all useful in categorization, i.e., in identifying category members. This knowledge forms the context-independent structure of basic level concepts that underlies the use of these concepts in language in any context.

In addition, however, basic level concepts are connected to thematic associates (concepts related by events rather than by taxonomic similarity) and to other non-centrally related concepts (e.g., to attributes not used or useful in categorization). Thus, they are understood with respect to our knowledge and theories about all other connected concepts. The thematic associates and other non-centrally related concepts form the context-dependent structure associated with a concept, i.e., structure that is relevant in particular situations. For example, *bottle* is a thematic associate of *baby,* and *mortgage* is non-centrally related to *house.* Thus, each concept in our system consists of large amounts of interrelated knowledge; however, we use no high-level knowledge organization structures such as frames or schemas. Instead, we use this relatively unorganized, interrelated knowledge to construct "concepts" appropriate to the current task and context in working memory during discourse processing, i.e., to construct the appropriate frame or schema.

We understand basic level concepts such as *dog* and *skunk* not only in terms of appearances and affordances, but also with respect to our deeper theories about *animals.* Thus, our understanding of basic and and subordinate level concepts also depends upon our understanding of superordinate level concepts. In experimental studies, subjects frequently list few or no attributes for these concepts [Tversky & Hemenway 1984]. Thus, our knowledge about superordinate concepts is a deeper and less easily verbalized knowledge, involving underlying principles and theories about the world. The representation of superordinate level concepts is a current area of research for us. We believe that it is an extremely important area.

4. DISCOURSE PROCESSING: IMPLICIT FOCUSING.

The processing performed by our system will be illustrated by considering the issues and problems in comprehending references to implicitly evoked entities in discourse. The problem of implicit focusing is an important one for natural language understanding systems, since references to entities not previously explicitly mentioned occur frequently in discourse. As stated previously, basic level concepts evoke a rich set of entities which may be referred to later in the discourse. Superordinate level concepts, in contrast, evoke few discourse entities.

4.1. Focusing Mechanisms. We have extended the grammar of SNePS/CASSIE [Shapiro 1982, Shapiro & Rapaport 1987] to handle implicitly focused definite anaphors. Resolution of references to explicitly focused entities makes use of a primary focus space; i.e., referents or co-specifiers are sought among the active elements of this focus space. Implicitly focused entities, however, require a second focusing mechanism, which makes use of an additional data structure called a *potential focus list* and the SNePS path-based inference package [Shapiro 1978].

4.2. Potential Focus List. When a new individual identified by its basic level name (e.g. a *dog*) or a generic basic level concept (e.g., the type *dogs*) is encountered in input, the context-independent satellite entities implicitly evoked by the central concept are placed in a potential focus list with the evoking concept. I.e., we believe that these reflex, or subconscious, inferences are made at the time of reading/hearing the central basic level concept. The kinds of inferences that are made are those discussed above: inferences about part-whole and other image-schematic structure, and inferences about perceptual structure. (Inferences about functional attributes are not currently activated, since the current discourse understanding task is concerned only with definite anaphora. Functional attributes are clearly a central component of basic level concept structure, however.) Inferences concerning context-dependent entities, i.e., thematic associates and other non-centrally related associates, are also made at this time.

When a subordinate level concept, e.g., a *Volkswagen*, is encountered, the reflex inference that *Volkswagens are cars* is made, and the subconscious inferences about *cars* are again drawn. Evidence suggesting that subordinate level concepts automatically activate their superior, basic level concepts has been provided by many researchers including Rosch [1978] and Barsalou [1982]. Thus, many of the subconscious inferences made when two different subordinates of the same basic level category are encountered are the same. This is supported by the inability of subjects to cite different properties of the sort discussed here, e.g., part attributes, to distinguish these subordinate categories [Tversky & Hemenway 1984; Rosch 1976]. E.g., *collies*, *poodles*, and *spaniels* share parts and differ on other types of attributes. Named individuals, such as *Lucy* (member of basic level category, *girl*) or *Rover* (member of basic level category, *dog*) are handled in the same way; i.e., the basic level concepts for these individuals are activated and the subconscious inferences about *girls* or *dogs* are made.

4.3. Path-Based Inference. The SNePS path-based inference package provides the subconscious reasoning that is required for implicit focusing. If a relation does not exist explicitly between two nodes in the network, one may specify a path of arcs which is semantically equivalent to the relation being sought. The definition of appropriate paths in the semantic network enables the automatic retrieval of the relevant satellite concepts of the basic level concepts: part structures, other image schematic structures, and other percepts.

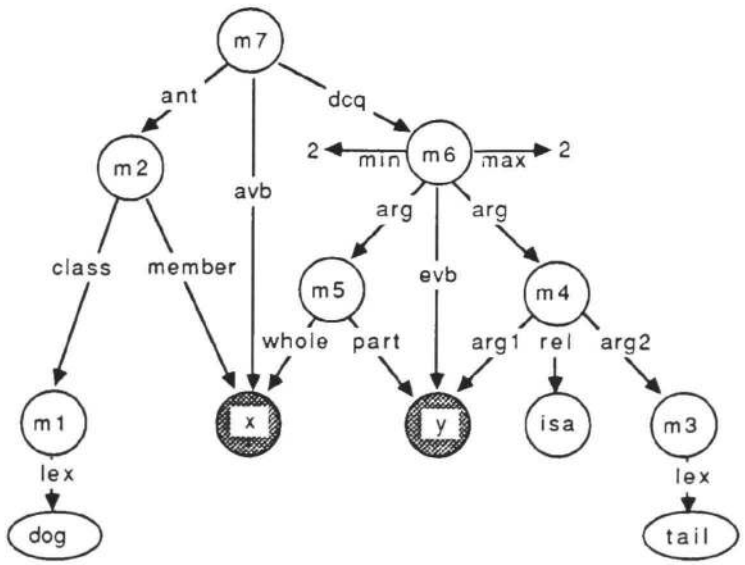
Thus, entities such as parts can be retrieved by defining a path of arcs from a node representing a basic level category, e.g., *dog*, to its parts: *nose, tail, head, ears, legs*, etc. In addition to the parts path, we can define paths of arcs to retrieve percepts (sounds, colors, smells) and the other image schematic structures. The ATN grammar then makes use of these defined paths to activate, i.e., implicitly focus, the context-independent satellite concepts of a basic level concept (that has been either encountered in input or activated by a subordinate level concept): returning all the nodes (concepts) that are found at the end of the defined paths of arcs emanating from the basic level concept and placing them in the potential focus list.

Since basic level categories are represented, in part, in SNePS using default generalizations or rules [Peters & Shapiro, 1987a], for every default rule in the network that says that members of the basic level category typically or presumably have part P, P is put into the potential focus list. Figure 1 shows the default rule that can be paraphrased as 'For all x if x is a dog then typically x has a tail'' or more simply as ''Typically, dogs have tails''. It also shows a defined path called *parts* which is used to activate the sub-part entities of basic level concepts. These activated entities ''fade out'' of the secondary focus list when the evoking concept fades out of the primary focus space.

In a like manner, the context-dependent information may be implicitly activated. These concepts are not automatically activated whenever a basic level or subordinate level concept is encountered in input, however. Rather, their activation is triggered in response to a specific context. Currently, our implementation is limited to the activation of context-dependent information in response to discourse comprehension of events, such as *buying/selling a car/house, renting a car/house, driving a car, walking/washing the dog.*

4.4. Timing of Activation. Empirical evidence that reflex inferences about context-independent satellite entities and context-dependent thematic associates are made at the time of comprehension of concepts is provided by numerous studies, including work by Barsalou & Ross [1986], and Walker & Yekovich [1986]. In addition, Walker and Yekovich have demonstrated that it is not the case that all of the information associated with a concept is activated during comprehension of the concept. Rather, only the centrally related concepts become part of the discourse model in working memory, and, thus, are available as antecedents; peripherally related concepts are not available, i.e., are not implicitly activated. In our terminology, centrally related concepts are the context-independent entities and additional entities evoked by the current context; peripherally related concepts are the associates of a generic concept in long-term memory that do not become activated in the current context.

4.5. General Focusing Mechanism for Definite Anaphora Comprehension. When a definite anaphor is encountered, both the primary focus space containing explicitly mentioned entities, and the potential focus list containing implicitly evoked entities may be searched for a co-specifier, and the discourse model updated appropriately. I.e., if the co-specifier is found in the potential focus list, it is now moved to the primary focus space with the appropriate level of activatedness; if the co-specifier is found in the primary focus space, its level of activatedness is simply updated. If a co-specifier is not found in either the primary focus space or the potential focus list, then knowledge-based processing, i.e., inferencing using the total knowledge base, is necessary.



The following defines a path to find all the parts of basic level concepts

(def-path parts (compose arg2- arg1 part- whole avb- ant class))

Figure 1

(1) :Lucy bought a Victorian house.
I understand that Lucy bought a Victorian house
exec: 11.63 sec

(2) :The mortgage is high.
I understand that the mortgage of the Victorian house is high
exec: 8.90 sec

(3) :John visited her.
I understand that John visited Lucy
exec: 9.38 sec

(4) :The butler opened the door.
I understand that a butler opened the door of the Victorian house
exec: 17:45 sec

(5) :The house is huge.
I understand that the Victorian house is huge
exec: 7.30 sec

(6) :Lucy bought a dog.
I understand that Lucy bought a dog
exec: 9.35 sec

(7) :Her barking wakes Lucy.
I understand that the barking of the dog is waking Lucy
exec: 11.45 sec

(8) :John owns a cat.
I understand that John owns a cat
exec: 9.38 sec

(9) :He bought a canary.
I understand that John bought a canary
exec: 9.65 sec

(10) :The canary is named Tweety.
I understand that Tweety is the canary
exec: 6.40 sec

(11) :The cat stalks Tweety.
I understand that the cat is stalking Tweety
exec: 8.53 sec

(12) :His tail is swishing.
I understand that the tail of the cat is swishing
exec: 8.81 sec

(13) :His chirp alerted John.
I understand that the chirp of Tweety alerted John
exec: 11.61 sec

(14) :Lucy walked the dog.
I understand that Lucy walked the dog
exec: 11.56 sec

(15) :The leash became tangled.
I understand that the leash of the dog became tangled
exec: 10.86 sec

Figure 2

5. DEMONSTRATION.

Figure 2, a short dialogue from a sample run of SNePS/CASSIE, illustrates some of our current capabilities. User input is on lines with the **:-prompt**; the system's output and timing information are on the lines that follow. The following comments highlight some of the important features of this dialogue.

In sentence 1, comprehension of the basic level concept *house* implicitly evokes many entities, including parts such as *the roof, windows,* and *doors*. This forms part of the context-independent structure of the concept *house*. In addition, the verb *bought* in conjunction with *house*, activates such context-dependent entities as the *mortgage* and *cost* of the *house*. Thus, the concept *house* constructed in working memory has been tailored to the current context.

Sentence 2 contains a reference to an implicitly focused item, *the mortgage* which was activated in response to the context of *buying a house*. Sentence 4 contains a reference to *the butler*, an entity that was not activated by either the concept *house* or the current context. Thus, comprehension of *the butler* requires inferencing using the knowledge base. A comparison of the timing information for sentences 2 and 4 illustrates that the comprehension time for a non-activated entity *(the butler)* is longer than that for an activated entity *(the mortgage)*.

Sentence 6 contains the basic level concept *dog*, which implicitly activates many context-independent entities, such as parts and other percepts. It also contains the event of *buying a dog*, so *the cost* is also activated. In sentence 7, 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 the potential focus list for the concept of *barking*, returning its evoking concept *dog*.

Sentences 12 and 13 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 a part of the basic level concept *cat* (in sentence 7), but not as a part of the subordinate level concept *canary;* whereas *chirp* was evoked as associated with *bird/canary* (in sentence 8), but not with *cat*. Our processing of *his tail* and *his chirp* simply involves searching the potential focus list for these previously activated entities, not a search of the knowledge base. Since the potential focus list also contains the evoking concepts, integration of the activated associates (e.g., *chirp*) with the previously mentioned evoking concept (e.g., *bird/canary/Tweety*) is quite simple. Finally, in sentence 14, *walking the dog* implicitly activates many context-independent entities, as well as context-dependent associates such as a *leash*.

6. CONCLUSIONS.

We have presented representations for natural language processing that are based on (1) a recognition of the unique nature of basic level categories within natural category systems, and (2) the need for "active", flexible representations for generic concepts and categories. Thus, we have based our system on principles of categorization derived from current categorization research. We believe that most AI/NLP systems fail to take these principles seriously, and are, therefore, unable to support natural language processing in anything other than extremely limited domains.

REFERENCES

(1) Barsalou, L. W. (1982), "Context-independent and context-dependent information in concepts," *Memory & Cognition*, vol 10, pp. 82-93.

(2) Barsalou, L. W., & Ross, B. H. (1986), "The Roles of Automatic and Strategic Processing in Sensitivity to Superordinate and Property Frequency," *Journal of Experimental Psychology*, vol 12, pp. 116-134.

(3) Barsalou, L. W. (1987), "The Instability of Graded Structure," In U. Neisser (ed.), *Concepts and Conceptual Development* (Cambridge: Cambridge University Press).

(4) Berlin, B. (1978), "Ethnobiological Classification," *Cognition and Categorization* (Hillsdale, NJ: Lawrence Erlbaum Associates), pp. 9-27.

(5) Bobrow, D. G., & Winograd, T. (1977a), "Experience with KRL-0, One Cycle of a Knowledge Representation Language," *IJCAI-77*, vol. 1, pp. 213-222.

(6) Bobrow, D. G., & Winograd, T. (1977b), "An Overview of KRL, a Knowledge Representation Language," *Cognitive Science*, vol. 1, pp. 3-46.

(7) Bobrow, D. G., & Winograd, T. (1979), "KRL: Another Perspective," *Cognitive Science*, vol. 3, pp. 29-42.

(8) Brachman, R. J., Fikes R. E., & Levesque, H. J. (1983), "KRYPTON: A Functional Approach to Knowledge Representation," *IEEE Computer*, vol. 16, pp. 67-73.

(9) Fahlman, S. E., (1979), *NETL: A System for Representing and Using Real-World Knowledge* (Cambridge, MA: MIT Press).

(10) Kintsch, W., & Mannes, S. M. (1987), "Generating Scripts from Memory," *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, Seattle, WA (Hillsdale, NJ: Lawrence Erlbaum), pp. 861-872.

(11) Lakoff, G. (1987), *Women, Fire, and Dangerous Things* (Chicago: The University of Chicago Press).

(12) Marr, D. (1982), *Vision* (San Francisco: Freeman).

(13) Mervis, C. B., & Rosch, E. (1981), "Categorization of Natural Objects," *Ann. Rev. Psychol.*, vol. 32, pp. 89-115.

(14) Minsky, M. (1975), "A Framework for Representing Knowledge," In P. H. Winston, (ed.), *Psychology of Computer Vision* (McGraw-Hill).

(15) Minsky, M. (1986) *The Society of Mind* (New York: Simon & Schuster).

(16) Murphy, G. L., & Medin, D. L. (1985), "The Role of Theories in Conceptual Coherence," *Psychological Review*, vol. 92, pp. 289-316.

(17) Neisser, U. (1987), "From Direct Perception to Conceptual Structure," In U. Neisser, (ed.), *Concepts and Conceptual Development* (Cambridge: Cambridge University Press).

(18) Peters, S. L., & Shapiro, S. C. (1987a), "A Representation for Natural Category Systems," *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, Seattle, WA (Hillsdale, NJ: Lawrence Ehrlbaum), pp. 379-390.

(19) Peters, S. L., & Shapiro, S. C. (1987b), "A Representation for Natural Category Systems," *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, Milan (Los Altos, CA: Morgan Kaufman), pp. 140-146.

(20) 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.

(21) Rosch, E., & Lloyd, B. B., (eds.), (1978), *Cognition and Categorization* (Hillsdale, NJ: Lawrence Erlbaum Associates).

(22) Schank, R. C., & Abelson, R. P. (1977), *Scripts, Plans, Goals, and Understanding* (Hillsdale, NJ: Lawrence Ehrlbaum Associates).

(23) Schank, R. C. (1985), *Dynamic Memory* (Cambridge: Cambridge University Press).

(24) 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.

(25) Shapiro, S. C. (1979), "The SNePS Semantic Network Processing System," In N.V. Findler, (ed.), *Associative Networks* (New York: Academic Press), pp. 179-203.

(26) Shapiro, S. C. (1982), "Generalized Augmented Transition Network Grammars for Generation from Semantic Networks," *American J. Computational Linguistics*, vol. 8, pp. 12-25.

(27) Shapiro, Stuart C., & Rapaport, William 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.

(28) Tversky, B., & Hemenway, K. (1984), "Objects, Parts, and Categories," *Journal Of Experimental Psychology: General*, vol. 113, pp. 169-93.

(29) van Dijk, T. A., & Kintsch, W. (1983), *Strategies of Discourse Comprehension* (New York: Academic Press).

(30) Walker, C. H., & Yekovich, F. R., (1987), "Activation and Use of Script-Based Antecedents in Anaphoric Reference," *Journal of Memory and Language*, vol. 26, pp. 673-691.

THE INDUCTION OF MENTAL STRUCTURES WHILE LEARNING TO USE SYMBOLIC SYSTEMS

T. G. Bever and Ralph E. Hansen
University of Rochester

ABSTRACT

Subjects learned to map phrase-structure-defined strings onto geometric figure arrays. "String-generation" subjects produced symbol strings corresponding to arrays; "String-interpretation" subjects constructed arrays corresponding to strings. "Mixed" subjects alternated between these tasks. Subjects' knowledge of symbol sequence acceptability was periodically probed. Mixed subjects learned the structure dramatically faster than other subjects. This suggests that natural acquisition of structure underlying symbol-world mapping systems like language depends on learning multi-directional mappings.

* * * * * * * *

Humans inevitably construct implicit mental representations to guide their concrete actions and percepts. The research in this paper investigates a theory of the conditions which elicit such formal mental structures. The essence of the theory is that these structures are mental devices which integrate superficially distinct perspectives on a situation. Consider the intuitive formation of a map between home and work. If you always walk from home to work, but get a ride home after dark, there is little basis for developing an intuitive map of the relevant neighborhood. Rather, you can memorize a unilinear series of turns and distances. Contrast this with a situation in which you walk sometimes in each direction: it is intuitively clear (though the relevant research remains largely undone), that you are then much more likely to construct an intuitive map of the area in an abstract representation of the streets and crucial landmarks. The differing perspectives gained from walking in both directions stimulate the instinctive need for a mental representation which is neutral concerning the direction of travel - namely, a map.

The functional value of such inner mental representations is unquestionable, but the fact that the structures are often functional does not explain their existence, form, or the dynamics of their discovery. In fact, it is frequently the case that humans ascribe unnecessarily elaborate internal structures to superficially regular phenomena. For example, people often intuitively invoke a complex causal schema as the internal structure of a series of events, which might in fact be unrelated; similarly, people may develop overly elaborate hypotheses about the structure of machines, as they learn to control them; finally, people acquire complex grammars with independent interlocking levels of representation to account for the structure of sentences, which might have simpler behavioral descriptions. In each case, the more superficial analysis might be correct or at least more functional - the person who attributes an unnecessary underlying structure is guilty of a cognitive illusion, the intuitive formation of an incorrect mental representation. As in the study of perception, the importance of such illusions is that they demonstrate the presence of an active set of mental processes which automatically form mental representations during the organization of behavior, regardless of their specific functional role. The

puzzle about why people induce complex abstract structures resolves into several component questions:

(1) Why do humans hypothesize the particular structures they do?
(2) What environmental conditions elicit the structures?
(3) What mental conditions elicit the structures?
(4) What motivates invoking the structure without direct reinforcement?

Questions (1) and (4) are usually taken to be the most profound: the former bears on hypotheses about innate constraints, the latter on the motives for active learning of abstract structures. It is difficult, however, to answer either question without a better understanding of the dynamics of the learning process. Accordingly, our research concentrates on the second and third questions: our theoretical goal is to understand the interactions between the environment and the learner's mental state which result in the formation of mental structures. Our practical goal is to develop some insights into the conditions that best elicit spontaneous formation of an appropriate mental representation for a situation.

Our theory of structure induction is rooted in the dynamic role of abstract representational schemata, as systems that resolve inconsistencies between superficial systems of representation (Bever, 1986). On this view, structure induction has some formal similarities to problem solving. It involves several phases; first, the formulation of distinct representations which seem to be inconsistent (the real mental 'problem'); then evocation of a more abstract representational schema which allows for the integration of the conflicting representations. An example of this is Duncker's (1945) classic explication of the solution for the use of x-rays to kill an internal tumor: at first, the subjects oscillate between postulating an x-ray 'gun' which shoots the tumor (but destroys the intervening tissue), and an x-ray 'bomb' which explodes only at the tumor site (but cannot get there because of the intervening tissue). The solution lies in an integration of features of both the 'gun' and the 'bomb'; a focussing lens disperses the x-rays, like light, harmlessly through the intervening tissue, and focusses them lethally only on the tumor. The concept of a lens which manipulates x-rays as though they were light provides a new schema in which to integrate the initially inconsistent representations of the problem.

On our view, the induction of structures underlying behavior works most effectively in an analogous way. Different superficial regularities, or different modes of use, stimulate the development of incompatible representations of the behavior: the deeper representation supplies a resolution of these apparently conflicting representations. This sequence of mental stages is a standard account of how children go about discovering elaborate mental systems, such as causal reasoning, number, naive physics, and so on. We are suggesting that the same kind of processes occur during adult learning of complex systems.

The use of symbol manipulation paradigms

Symbol-sequence learning offers a rich paradigm to examine the induction of an abstract structure from specific concrete training experiences. In these experiments, subjects typically are asked to discover the principles

underlying the well-formedness of sequences. The paradigm offers the possibility of experimental control over the intermediate stages of structure formation, and careful probing for the ultimate structures.

In practice, such studies are often described as investigations of 'artificial language learning', since the description of the symbol sequences is often expressed in terms of language-like rules (Anderson, 1975; Braine, 1963, 1966; Esper, 1925; Miller, 1967; Miller and Stein, 1963; Moeser & Bregman, 1972; Reber, 1967; Saporta, Blumenthal and Reiff, 1963; Segal and Halwes, 1965, 1966; Smith, 1969). The rationale for these studies has usually been taken to be that one can include or violate formal properties of natural language in the artificial mapping systems; if the selective absence of a particular formal property makes the language hard to learn, then one might conclude that the property is a critical part of the structure of any language (Esper, 1925; Chomsky, 1965). That is, artificial languages can be used to test linguistic universals, one by one.

Such learning paradigms have been used to investigate some behavioral issues, primarily contrasting the importance of structural information about where phrases begin and end (Green, 1979; Morgan and Newport, 1981; Morgan, Meier and Newport, 1986) and the relative importance of parallels between grammatical structure and its extra-symbolic reference for learning (Moeser and Bregman, 1972, 1973; Moeser, 1977; Anderson, 1975; Meier and Bower, 1986).

These studies of symbol sequence learning have had the same limitations as other paradigms one might use to explore naturalistic discovery of structure. They pose subjects directly with the problem of discovering the structures at issue, rather than placing subjects in tasks which might naturally elicit the structures as implicit components of the solution to the tasks. Below, we outline an adaptation of artificial language learning paradigms in which subjects learn to <u>use</u> the symbolic structure without being asked to learn the structure <u>explicitly</u>. Our initial results suggest that the paradigm can be used to answer questions about the behavioral conditions governing the discovery of implicit structures in general.

THE STUDY

In our study, subjects learn to use symbols in sequences defined by simple structural constraints, and are not given any direct training on structural well-formedness. As they learn to use the symbol-sequences, we periodically test their knowledge of the structure of the symbol system. Subjects are either asked to learn to 'produce' symbol sequences correctly or to 'perceive' them. In 'production', the subjects are given a visual array of shapes on a computer screen and must type the sequence describing the array. In 'perception', subjects are given a sequence and must construct the visual array which it describes.

The symbol sequences are structured according to rules taken from a standard artificial language used in previous studies in the literature (e.g., Anderson, 1975; Meier and Bower, 1986). It is a phrase structure verb-final language, with embedding. Sequences ranged from 4 to 12 words long; the separate words are in English ('triangle, red, large, above' etc.) except for one grammatical function word, 'te'. The visual pattern is

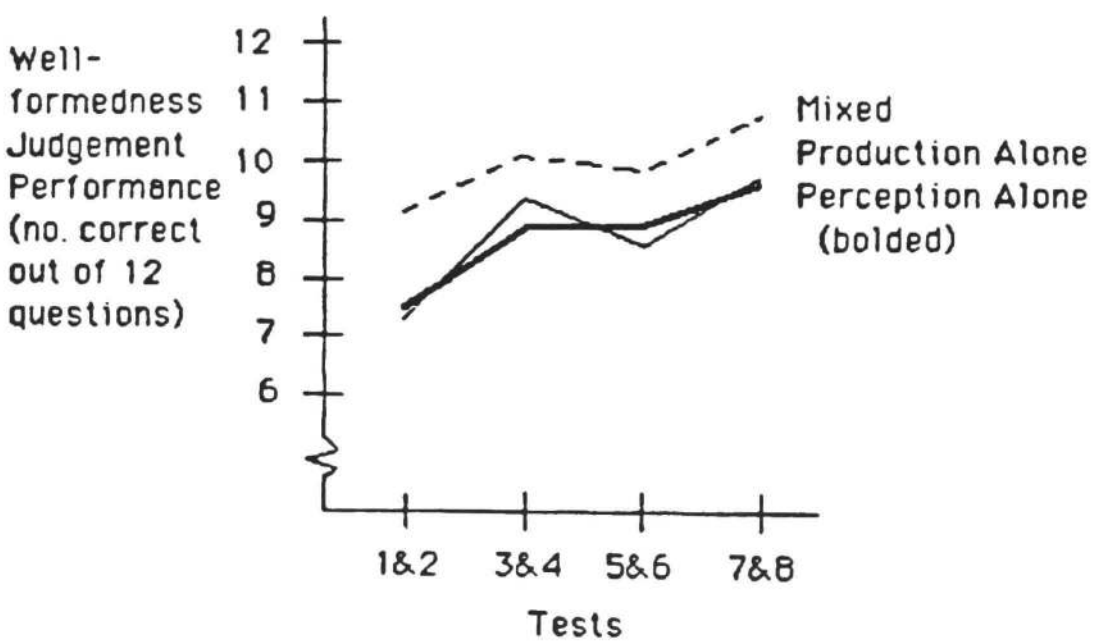
defined on a set of geometric shapes which can have different sizes and colors. Each shape can be located in one of four quadrants on a computer screen. Subjects can 'paint' the figure they want in each quadrant separately, using an adapted graphics package: they are provided with labelled buttons for each figure attribute (triangle, red, large, etc.). Even this simple mapping system allows for complex mappings. For example, a large striped triangle above a red circle, which is to the left of a small striped square, would be denoted by the string,

triangle large striped circle te red te square small striped left-of above

The subjects are never trained on isolated sequences; rather, they are exposed to unidirectional mapping tasks which naturally reflect the normal uses for symbol systems, comprehension and production. In all conditions, subjects are pre-trained in mapping isolated symbols, to become familiar with using the computer-controlled printing and drawing techniques.

We ran groups of 10 subjects, balanced for such variables as SAT scores, sex and age, in each of three paradigms: 1) 'perception': on each trial, subjects are given a symbol sequence and asked to construct the corresponding visual pattern; 2) 'production': on each trial, subjects are given the visual pattern and asked to construct a corresponding symbol sequence; and 3) 'mixed'; trials alternate between 'perception' and 'production'. There were 48 trials, selected to balance for various complexity variables in each 1/8 of the experimental session. After every 6 trials, subjects were presented with a test of their knowledge of the structure of well-formed sequences. Subjects judged which member of each of six pairs of sequences is structurally correct. Following previous research, the correct sequence in each pair was mated to an incorrect sequence which violated one of six kinds of structural properties characteristic of the system.

FIGURE 1



There are two measures of performance: the acquisition of structural knowledge, and the acquisition of the unidirectional mapping skills. Figure 1 shows how the structural knowledge increased with training in the three different training conditions. Training in production resulted in slightly faster acquisition of structural knowledge than did training in perception, but the difference is not significant. Most striking is the fact that the mixed condition resulted in superior mastery of the structure of the symbol system ($p<.025$, by Fisher exact test on subjects, both comparing mixed against perception alone and against production alone ($p<.03$ by a Wilcoxon matched-pairs signed-ranks across trials). This finding is not obvious: for example, one might have predicted that structural learning in the mixed condition would be the average of that in the two separate conditions. Furthermore, correlations of subjects' performance on the second half of the session shows that structural knowledge correlated strongly with production and perception in the mixed condition, but less strongly with production or perception alone (see Table 1, below). It seems clear that the mixed condition elicited a more unified representation of the structure with the behavioral skill.

TABLE 1

Correlations across subjects between mapping skill and well-formedness judgments (second half of sessions).

Production alone X judgment	.64 ($p<.02$)
Production mixed X judgment	.93 ($p<.001$)
Perception alone X judgment	.61 ($p<.03$)
Perception mixed X judgment	.80 ($p<.001$)

FIGURE 2

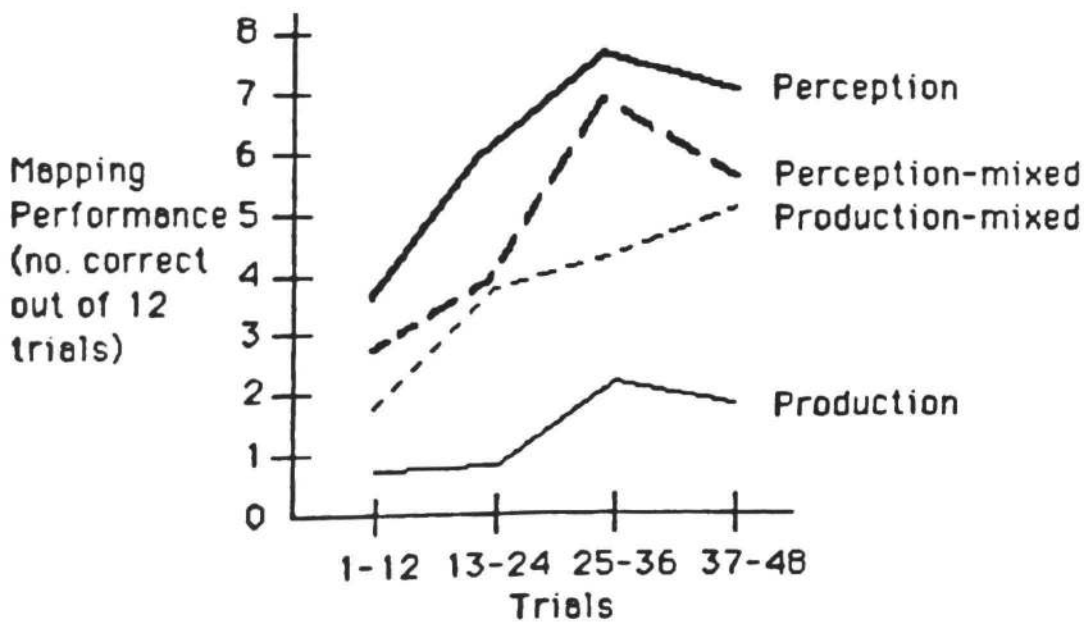


Figure 2 presents the acquisition of just the mapping skills in the different conditions. Production alone is clearly more difficult to master than perception alone; but production in the mixed condition is almost as easy as perception (we took performance on the last 12 trials as a measure: on this, production alone was more difficult than each of the other three conditions, $p<.025$, by a Fisher exact test across subjects; $p<.02$ by a Wilcoxon matched-pairs signed-ranks test on trials). Finally, perception in the mixed condition is no easier than perception alone. Several aspects of the results suggest that the acquisition of production is more directly related to the ability to make structural judgments than the acquisition of perception. First, the correlation across subjects between mapping performance and structural judgments is higher for production than perception in the mixed condition; second, the correlation across structural properties in structural judgement is higher for production and mixed, than for perception and mixed. These trends were not statistically significant given the current number of subjects and constraints, but they are suggestive as the basis for further research.

The initial findings from this study have a number of implications. First, the fact that structural knowledge is arrived at much more quickly when learning to map in both directions suggests that structural knowledge may be discovered as an integrated solution to multiple representational constraints. This follows as a special case of our original hypothesis that structure induction is facilitated if it provides a framework for incompatible systems of representation - clearly, perception and production in our paradigm involve distinct input/output relations. This contrast is emphasized by the fact that structural properties that are hard to master in one mode tend to be easier in the other. The mixed condition may be effective for independent reasons, for example, because the conflicting generalizations are in different modalities. The study of this has exciting implications for theories of the induction of structure not just of toy rule systems, but of such mental abstractions as causal reasoning, and such concrete objects as complex machines, computer algorithms and so on.

References

Anderson, J.R. (1975). Computer simulation of a language acquisition system: A first report. In R. L. Solso (Ed.), Information processing and cognition: The Loyola Symposium (pp. 295-349).

Bever, T.B. (1987). The aesthetic constraint on cognitive structures. In M. Brand and R. Harnish (Eds.), The representation of knowledge and belief. Tuscon, AZ: University of Arizona Press.

Bever, T.G., Fodor, J.A., & Weksel, W. (1965). On the acquisition of syntax: A critique of "contextual generalization." Psychological Review, 72, 467-482.

Braine, M.D.S. (1963). On learning the grammatical order of words. Psychological Review, 70, 323-348.

Chomsky, N. (1965). Aspects of the theory of syntax. Cambridge, MA: MIT Press.

Duncker, K. (1945). On problem-soilving (translated by L.S. Lees). Psychological Monographs, 58, No. 270.

Esper, E.A. (1925). A technique for the experimental investigation of

associative interference in artificial linguistic material. Language Monographs, No. 1.

Green, T.R.G. (1979). The necessity of syntax markers: Two experiments with artificial languages. Journal of Verbal Learning and Verbal Behavior, 18, 481-496.

Lenneberg, E.H. (1967). Biological foundations of language. New York, NY: Wiley.

Meier, R.P., & Bower, G.H. (1986). Semantic reference in phrasal grouping in the acquisition of a miniature phrase structure language. Journal of Memory and Language.

Miller, G.A., & Stein, M. (1963). Grammarama I: Preliminary studies and analysis of protocols. Scientific Report #CS-2, Harvard University Center for Cognitive Studies.

Miller, G.A. (1967). The psychology of communication. New York, NY: Basic Books

Moeser, S.D. (1977). Semantics and miniature artificial languages. In J. Macnamara (Ed.), Language learning and thought. New York, NY: Academic Press.

Moeser, S.D., & Bregman, A.S. (1972). The role of reference in the acquisition of a miniature artificial language. Journal of Verbal Learning and Verbal Behavior, 11, 759-769.

Morgan, J.L., Meier, R.P., and Newport, E.L. (1986). Structural packaging in the input to language learning: Contributions of intonational and morphological marking of phrases to the acquisition of language. (Under Review)

Morgan, J.L., & Newport, E.L. (1981). The role of constituent structure in the induction of an artificial language. Journal of Verbal Learning and Verbal Behavior, 20, 67-85.

Newport, E.L. (1982). Task specificity in language learning? Evidence from speech perception and American Sign Language. In E. Wanner and L.R. Gleitman (Eds.), Lanaguage acquisition: The state of the art (pp. 450-486). Cambridge: Cambridge University Press.

Reber, A.S. (1967). Implicit learning of artificial grammars. Journal of Verbal Learning and Verbal Behavior, 5, 855-863.

Saporta, S., Blumenthal, A.L., & Reiff, D.G. (1963). Grammatical models and language learning. In R.J. DiPietro (Ed.), Report of the Fourteenth Annual Round Table Meeting on Linguistics and Language Studies, (pp. 133-142). Georgetown University Monograph Series on Languages and Linguistics.

Segal, E.M., & Halwes, T.G. (1966). The influence of frequency of exposure on the learning of phrase structure grammar. Psychonomic Science, 4, 157-158.

Smith, K.H. (1969). Learning co-occurrence restrictions: Rule learning or rote learning? Journal of Verbal Learning and Verbal Behavior, 8, 319-321.

TRANSITORY STAGES IN THE DEVELOPMENT OF MEDICAL EXPERTISE: THE "INTERMEDIATE EFFECT" IN CLINICAL CASE REPRESENTATION STUDIES[1]

Henk G. Schmidt, Henny P. A. Boshuizen & Pie P. M. Hobus

University of Limburg, Maastricht, The Netherlands

One of the best-established empirical phenomena in research on the development of expertise in medicine, is the "intermediate effect " in clinical case representation studies. The experimental paradigm which produces this phenomenon is described as follows: Subjects differing in level of expertise are requested to study, for about two or three minutes, half a page of text describing a patient's history, presenting complaints and some additional findings--results of laboratory tests and/or physical examination. The text is removed, and the subjects are asked to recall everything they can remember from the text[2]. In addition, they are required to state a most likely diagnosis for the case. Subjects with intermediate levels of expertise consistently produce more extensive recalls than either experts --e.g. experienced physicians-- or novices (Patel & Groen, 1986a).
This phenomenon has been demonstrated under various conditions, with different cases and in different populations. In a study using a cardiovascular respiration case, Muzzin, Norman, Feightner, & Tugwell (1983) found that internists recalled significantly fewer items from the case than did residents or pre-residency medical students. Claessen & Boshuizen (1985) presented typical and atypical cases of pancreatitis and prostatitis to family physicians and students at three levels of expertise. The residents participating in the experiment showed superior recall on all cases as compared with both the family physicians and the pre-clinical students. Patel & Medley-Mark (1985) demonstrated the same phenomenon in final year medical students as compared with both internists and novices, using an acute bacterial endocarditis case and a stomach cancer case. The intermediate effect has, in addition, been demonstrated in expertise-related tasks other than text processing (Grant & Marsden, 1988; Patel, Evans, & Kaufman, 1988). In all studies reviewed however, diagnostic accuracy of the most experienced group exceeded that of the other levels of expertise, suggesting that the intermediate effect cannot be explained away by assuming lack of ecological validity of the experimental tasks.

These results appear to be counterintuitive. Research into the way in which subjects with different levels of prior knowledge process text has shown that the richer the knowledge base used in the interpretation of new information, the better recall of that information. Spilich, Vesonder & Voss (1979), for instance, found that subjects with high knowledge of baseball remembered more, and more relevant information from a report of a baseball game than low-knowledge individuals. In the domain of chess, grandmasters were able to recall positions of pieces on the board more accurately than players with less experience (De Groot, 1946). It is generally assumed that prior knowledge provides scaffolding for the new information to be encoded and retrieved. In other words: Instead of the invertedly U-shaped curve commonly found in developmental studies in medicine, one would expect a monotonous increase of recall as a function of increasing expertise.

---

[1]Preparation of this paper was enabled by grant # 6266 of the Dutch Foundation for Educational Research (SVO) to H. G. Schmidt and H.P.A. Boshuizen. The authors like to acknowledge the contributions of Steve Forster and Lorence Coughlin.

[2]The resulting recall protocol is segmented into propositions. The propositions recalled, and their interrelations, are considered the subject's mental representation of the patient's problem, a representation on which his diagnosis of the underlying pathology is based. One simple way to deal with a recall protocol is to count the number of propositions correctly recalled.

Based on an extensive review of the literature, Schmidt & Norman (1988) suggest that the intermediate effect may result from the fact that experts and intermediates apply different knowledge in the representation of a clinical case. In their view, the development of expertise in medicine progresses through several transitory stages, each of which is characterized by functionally different knowledge structures underlying performance. When applied to the understanding of clinical cases, these structures produce quite different effects. Novices, by their nature, have little more than a lay-person's idea of illness. Their knowledge is limited and consists mainly of an understanding of basic biological processes and structures, without much reference to the consequences of disease as exemplified in a clinical case. Students however, who enter residency, have already developed rich and elaborated causal networks explaining the signs and symptoms associated with a disease in terms of underlying pathophysiological processes, principles or mechanisms. Since, by that time, their exposure to "real" patients still has been limited, they have to process information extracted from a new case consciously and elaborately, reasoning through the causal pathophysiological networks available to them, in order to arrive at an understanding of that case. This reasoning process may take (considerable) time.
For experienced physicians however, causal pathophysiological knowledge has become compiled into diagnostic labels or simplified causal models explaining signs and symptoms, as a result of extensive use. Compiled knowledge is, by its nature, automatically and effortlessly activated by relevant cues in a case, because repeated activation in response to these same cues has caused its compilation (Anderson, 1983). There is some evidence supporting the idea of the limited use of elaborate pathophysiological explanation by experienced clinicians. Boshuizen, Schmidt & Coughlin (1987), using a think-aloud methodology, showed that general practitioners only rarely refer to pathophysiology while reasoning about a sequentially presented case. Whereas students use pathophysiological concepts extensively when explaining the information presented to them. Patel, Evans & Groen (1988), reviewing research on pathophysiological explanation of clinical cases by subjects of different expertise, also conclude that experts appear to rely less on causal biomedical knowledge while diagnosing a case. These findings support the idea that the development of expertise involves extensive compilation of knowledge acquired during training. However, Schmidt & Norman postulate that, in addition to the process of compilation, another process takes place as a result of frequent exposure to patients. Experience adds something to the knowledge base of physicians that is only superficially taught in medical school but appears to be most relevant while diagnosing a patient: The constraints under which disease occurs in humans. There is some evidence that expert behavior is determined in particular by the extent to which rich and elaborated knowledge about these constraints is acquired (Schmidt, Hobus, Patel, & Boshuizen, 1987). To accommodate this point of view, Feltovich & Barrows (1984) have suggested that in the course of years of practice, physicians develop cognitive structures of various diseases which they call "illness scripts". These illness scripts contain the physician's idiosyncratic and compiled knowledge of the disease and its consequences, in addition to knowledge of the constraints under which a disease occurs. ("Enabling conditions" is the term they use to refer to these constraints.) Illness scripts are frame- or list-like structures, containing prototypical information about a disease, which, when activated, guide a clinician through a case and support him in looking for cues that are relevant.

This theory allows for a number of predictions about the nature of differences between subjects of various levels of expertise. Of these, the following are relevant to the experiment described here:
1. Assuming that conscious application of causal pathophysiological knowledge in explaining all the information embedded in a case can be considered more extensive than just looking for relevant cues in order to match an applicable illness script, it may be expected that the recall of that case by intermediates will be superior to the experts' recall. This would explain the source of the intermediate effects.
2. Conscious processing, however, takes more time than automatically activating a relevant script and filling in slots. Hence, it is predicted that the intermediate effect will disappear when processing time is restricted. The experts' recall performance however, will be less affected by a

decrease in the time available to process the case, simply because experts need less time to match the symptoms to a relevant script. The script in turn, will facilitate subsequent retrieval of case-related information.

3. If intermediates usually process a case activating elaborate pathophysiological knowledge, whereas the experts only apply compiled knowledge, it is predicted that post-hoc pathophysiological explanations provided by intermediates will be more extensive than those provided by experts.

4. Prediction 3 will only apply to those circumstances in which subjects have enough time to apply causal knowledge. Thus, a decrease in processing time available will also affect the elaborateness of the post-hoc pathophysiological explanations provided by intermediates. The amount of explanation provided by the experts however, will not be influenced by manipulation of the processing time, since their knowledge is compiled and easily available, even under highly restricted processing conditions.

5. Limitation of processing time will probably affect the accuracy of the diagnostic hypotheses offered by all groups. However, the expert group will display smaller losses in accuracy for the same reasons as stated before: Experts reason less and therefore, have more time available to scan briefly the case in search of a solution.

In order to test these hypotheses, subjects having different levels of expertise were requested to study a case history under varying time constraints, recall the information, provide a diagnosis and produce a pathophysiological explanation.

METHOD

Subjects

Subjects were 120 students and physicians of the University of Limburg: 24 first year allied health sciences students, 72 medical students (24 second-, 24 fourth- and 24 sixth-year undergraduate students) and 24 internists with at least two years of experience. Each group of 24 was randomly subdivided in three groups of eight who studied the clinical case under different time constraints. Subjects received a small compensation for their participation.

Material

The materials consisted of a booklet containing a 270-word description of a clinical case and three blank response sheets. The case was a Dutch translation of the acute bacterial endocarditis case used by Patel & Groen (1986b) and consisted of 71 propositions.

Procedure

First, subjects were requested to study the case carefully. Dependent upon the experimental condition, they were given the opportunity to study the case for 3 minutes and 30 seconds (3' 30"), which was the amount of time allotted to subjects in the original Patel studies, 1 minute 15 seconds, or 30 seconds[3]. After reading the text for the duration of time allowed, the experimenter asked them to turn to the next page. This page contained the following instruction: "Would you be so kind as to write down everything you recall from the case? Write complete sentences and avoid abbreviations". Sufficient space was given to produce a free recall protocol. On the next page, subjects were requested to provide a diagnosis for the patient. The last page contained this instruction: "Describe the pathophysiology which, in your opinion, underlies the case. Write complete sentences and avoid abbreviations. Only use schematic representations if strictly necessary". Subjects were free to use as much time as they needed for the assignments.

---

[3]Preceding the processing of the case, subjects in the 30" condition were given the opportunity to read an unrelated text of exactly the same length to provide them with some experience in scanning a text in a very short time. This was done in order to minimize variability in the way subjects would undertake the experimental task.

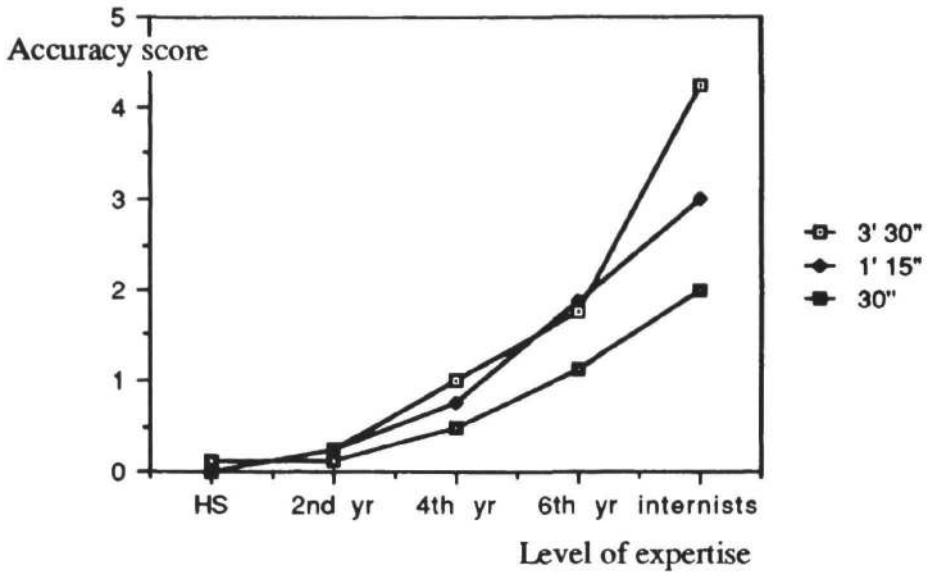


Figure 1. Average accuracy of diagnoses as a function of expertise and processing time

Subsequently, the free recall and pathophysiology protocols were segmented into propositions, applying a technique adapted from Frederiksen (1975). The number of propositions correctly recalled was recorded. In addition, the total number of propositions in the pathophysiology protocols was established. The accuracy of the diagnosis was determined attaching weights to each of the elements. If the diagnosis contained the term "endocarditis" 2 points were given. The presence of "acute", "bacterial" and "embolisms" each contributed 1 point. So, a maximum score of 5 could be obtained.
In addition, extensive qualitative analyses were carried out which are beyond the scope of this paper and whose results will not be discussed here.

RESULTS and DISCUSSION

Diagnostic accuracy

Figure 1 shows the average accuracy of the diagnoses proposed by the subjects. In line with other studies, the number of accurate diagnoses significantly covaries with expertise, resulting in the classical monotonically increasing performance curve. These data suggest that the case representation task is ecologically valid, because expertise-related differences to be expected are actually found.
Processing time, however, does not have a significant effect on diagnostic performance. This conclusion applies to all levels of expertise, including expert performance: $F(2, 21) = 2.62$, $p < .10$. These results are somewhat at variance with the expectation that, in particular, performance of subjects with less experience would suffer from the constraining of time available for processing. However, since the average scores of these groups are extremely low under all conditions, failure to find significant differences may be due to the overall difficulty of the case.

Recall data

Figure 2 shows the results of the analyses of the free recall protocols. The number of propositions recalled is displayed as a function of expertise and processing time. Overall differences between levels of expertise, and within different processing time conditions, are statistically significant ($p < .001$). An exception is the performance of the different groups under the intermediate --1' 15"-- processing time constraint; $F(4, 35) = 2.33$, $p < .07$. Pairwise comparisons within conditions show the following pattern: Under the 3' 30" processing time condition, the internists and the health sciences students produce significantly less propositions than the three intermediate levels of expertise ($p < .001$), nicely illustrating the existence of an intermediate effect in these data. Among

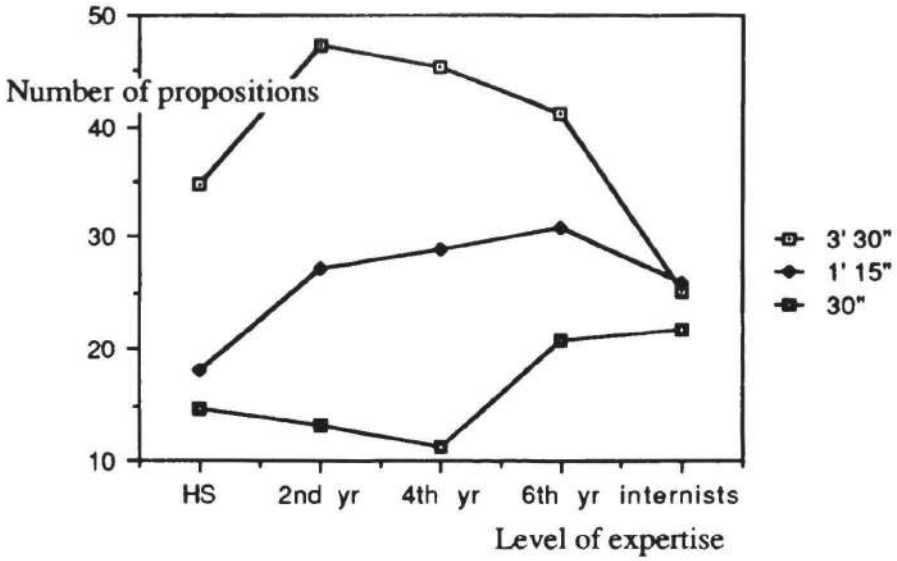


Figure 2. Number of propositions recalled from the acute bacterial endocarditis case as a function of expertise and processing time.

the groups that were required to process the case within 1 minute and 15 seconds, only the fourth- and sixth-year students produce significantly more recall, indicating that the intermediate effect, although still traceable, is vanishing. Under the shortest processing condition, the two highest levels of expertise differ significantly from the other groups, but not from each other.
In summary, as predicted by Schmidt & Norman (1988), the emergence of the intermediate effect appears to be dependent on the amount of time available for processing clinical information. Given a sufficiently short period of time, the phenomenon disappears, and is replaced by an increase in performance as a function of expertise. It is interesting to note that differences within levels of expertise as a result of differences in processing time are highly significant (all show p-values smaller than .0001), with the exception of the internists; $F(2, 21) = 1.04$, $p < .37$. These data suggest that in contrast to students of all levels of expertise, the performance of experienced physicians is relatively insensitive to manipulations of time, at least within the limits of the present experiment.

Pathophysiological protocols
The hypothesis was that intermediates and experts use functionally different knowledge while representing a clinical case. According to Schmidt & Norman (1988), intermediates consciously process the information applying a rich base of causal pathophysiological knowledge, whereas experts almost automatically activate compiled knowledge directly relevant to the case. Thus, post-hoc pathophysiological explanations provided by the intermediates will be more extensive than those provided by the experts. However, if intermediates are hindered in unfolding their knowledge, by restricting time needed to activate and reason from relevant pathophysiological networks, information from the case cannot properly be processed. Under the latter condition, the elaborateness of post-hoc explanation by intermediates is expected to decrease, whereas pathophysiological explanation provided by the experts --although compiled and therefore less extensive-- will remain constant.
Figure 3 contains quantitative information concerning this issue. Again, overall differences are highly reliable, both within levels of the experimental treatment and within each of the levels of expertise. Exceptions are the 6th year students; $F(2, 21) = 2.28$, $p < .13$; and the internists; $F(2, 21) = .40$, $p < .67$. Among students, a decrease in processing time generally causes a decrease in the number of propositions produced in the pathophysiological protocols, suggesting that less extensive processing of pathophysiological knowledge has taken place while trying to understand the case.

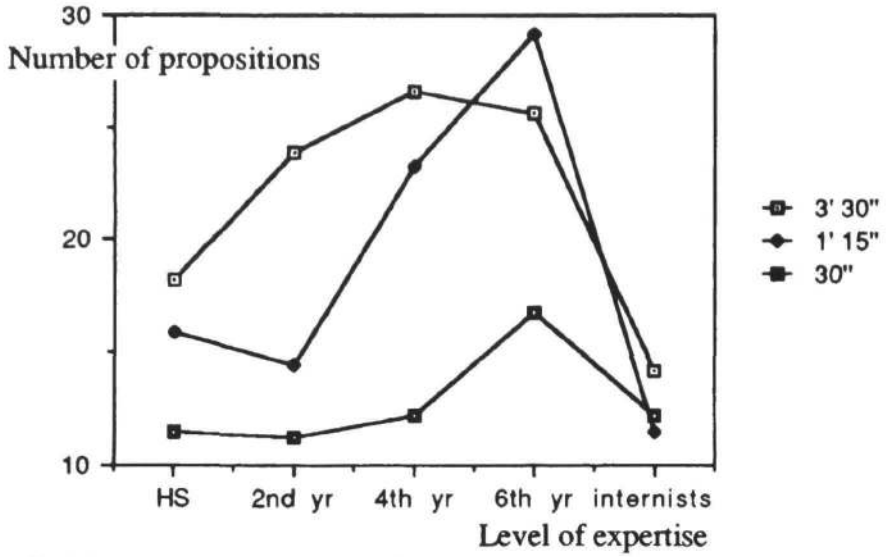


Figure 3. Number of propositions found in the pathophysiological protocols as a function of expertise and processing time

In many ways, these data resemble those of Figure 2. An intermediate effect is present, and processing time seems to have the same effect on pathophysiological reasoning as it has on recall. These similarities indeed suggest that the intermediate effect in free recall is the result of extensive and conscious processing of pathophysiological knowledge in the course of understanding the case. (Although the average within-group product-moment correlation between numbers of propositions produced in free recall and pathophysiology protocols is not impressive: $r = .34$, $p < .001$; largely due to the presence of zero correlations in two lowest expertise levels) Again, the performance of the physicians appears to be a separate issue. Their output is small and quite stable, irrespective of the constraints. Qualitative analysis shows that their protocols contain highly relevant high level explanatory concepts and compiled causal reasoning about the case. In addition, they incorporate in their explanations to a larger extent conditions that enabled the acute bacterial endocarditis to emerge (Boshuizen, 1988).

GENERAL DISCUSSION

The theory outlined rather simply explains a number of hitherto unexplained empirical phenomena, notably the intermediate effect repeatedly demonstrated in recall studies (Patel & Groen, 1986a; Muzzin, et al., 1983) and the relative absence of pathophysiological reasoning by expert-clinicians in think-aloud experiments (Boshuizen, et al, 1987; Patel, et al.,1988).

The data presented generally appear to support the notion proposed by Schmidt & Norman (1988) that students and experienced physicians represent clinical cases in different ways, because in the process of understanding the text, both groups use functionally different knowledge. Provided they have sufficient time, medical students consciously process causal pathophysiological knowledge activated by cues embedded in the text. Physicians, however, only pick up information that is directly relevant to the solution of the problem, because their knowledge of disease, its enabling conditions, causes and consequences, has been etched by continuous exposure to thousands of patients. The resultant structures are highly compiled in order to meet the demands of their profession.

Of course, the data presented only suggest a cause-effect relationship between the kind of knowledge used and recall performance. Further research is needed to establish the causal chain between the two. A particularly attractive option in this respect seems to be the experimental manipulation of prior knowledge of groups having different levels of expertise, in order to investigate effects of knowledge activation on the representation of clinical information in memory.

REFERENCES

Anderson, J. R. (1983). *The architecture of cognition*. Cambridge, MA: Harvard University Press.

Boshuizen, H. P. A. (1988). De rol van kennis in de ontwikkeling van expertise in de geneeskunde (The role of knowledge in the development of expertise in medicine). Doctoral dissertation. Maastricht, The Netherlands: University of Limburg Press.

Boshuizen, H. P.A., Schmidt, H.G., & Coughlin, L.D. (1987). *On-line representation of a clinical case and the development of expertise*. Paper presented at the Annual Meeting of the American Educational Research Association. Washington, DC.

Claessen, H.F.A. & Boshuizen, H.P.A. (1985). Recall of medical information by students and doctors. *Medical Education; 19*, 61-67.

De Groot, A. D. (1946). Het denken van den schaker (Thinking processes in chess players). Den Haag: North Holland.

Feltovich, P.J. & Barrows, H.S. (1984). Issues of generality in medical problem solving. In H.G. Schmidt & M.L. de Volder. (Eds.). *Tutorials in problem-based learning*. Assen: Van Gorcum.

Frederiksen, C.H. (1975). Representing logical and semantic structures of knowledge acquired from discourse. *Cognitive Psychology, 7*, 371-458.

Grant, J. & Marsden, P. (1988). Primary knowledge, medical education and consultant expertise. *Medical Education, 22*, 746-753.

Muzzin, L. J., Norman, G. R., Jacoby, L. L., Feightner, J. W., Tugwell, P., & Guyatt, G. H. (1982). Manifestations of expertise in recall of clinical protocols. *Proceedings of the 21th Annual Conference on Research in Medical Education* . Washington, DC.

Muzzin, L. J. , Norman, G. R. , Feightner, J. W., & Tugwell, P. (1983). Expertise in recall of clinical protocols in two specialty areas. *Proceedings of the 22nd Conference on Research in Medical Education*. Washington, DC.

Patel, V. L., Evans, D. A. , & Groen, G. J. (1988). Biomedical knowledge and clinical reasoning. In D. A. Evans & V. L. Patel. (Eds.), *Cognitive science in Medicine*. Cambridge, MA: MIT Press.

Patel, V. L., Evans, D. A., & Kaufman, D. (1988). A cognitive framework for doctor-patient interaction. In D. A. Evans & V. L. Patel. (Eds.), *Cognitive science in Medicine*. Cambridge, MA: MIT Press.

Patel, V. L. & Medley-Mark, V. (1985). Knowledge, integration from clinical texts, use of factual, inferential and integrative questions. *Proceedings of the 24th Annual Conference of Research in Medical Education*. Washington, DC.

Patel, V. L. & Groen, G. J. (1986a). Non-monotonicity in novice-intermediate-expert comparisons. *Proceedings of the 27th Annual Meeting of the Psychonomic Society*. New Orleans, LA.

Patel, V.L. & Groen, G.J. (1986b) Knowledge-based solution strategies in medical reasoning. *Cognitive Science, 10*, 91-116.

Schmidt, H. G., Hobus, P. P. M., Patel, V. L., & Boshuizen, H. P. A. (1987). *Contextual factors in the activation of first hypotheses: Expert-novice differences*. Paper presented at the Annual Meeting of the American Educational Research Association. Washington, DC.

Schmidt, H. G. & Norman, G. R. (1988). *On the development of expertise in medicine: Evidence from case-representation studies*. Technical Report #123. University of Limburg, Maastricht, The Netherlands.

Spilich, G. J., Vesonder, G. T., Chiesi, H. L., & Voss, J. F. (1979). Text processing of domain related information for individuals with high and low domain knowledge. *Journal of Verbal Learning and Verbal Behavior, 18*, 352-373.

Integrating Marker Passing and Connectionism for Handling Conceptual and Structural Ambiguities*

Ronald A. Sumida
Michael G. Dyer
Margot Flowers

AI Lab/Computer Science Dept.
University of California, Los Angeles, CA, 90024

Abstract

This paper discusses the problem of selecting the correct knowledge structures in parsing natural language texts which are conceptually and structurally ambiguous and require dynamic reinterpretation. An approach to this problem is presented which represents all knowledge structures in a uniform manner and which uses a constrained marker passing mechanism augmented with elements of connectionist models. This approach is shown to have the advantage of completely integrating all parsing processes, while maintaining a simple, domain-independent processing mechanism.

1. Introduction

A major problem in parsing natural language texts is the selection of the correct knowledge structures from the large number of inappropriate ones in memory. This problem is especially difficult in the case of texts which are highly ambiguous and which require the reader to correct an initially mistaken interpretation, since structures which are only potentially relevant must also be found. Consider, for example, the following sentence: **S1. John put the pot on the stove.** This seems to indicate that John is preparing to use a container for cooking on a stove. However, after reading the next sentence: **S2. He picked it up and smoked it**, it appears that John was actually using the stove as a supporter (or lighter) for a marijuana cigarette (not a cooking pot). In addition note that S1 and S2 are potentially ambiguous at the structural level, e.g. <X picked it up> could mean <X learned new information>, while <X put object on> could mean <X wear object>.

Previous approaches to parsing natural language texts have largely been unsuccessful at handling ambiguous sentences such as those presented above. These approaches can generally be divided into four groups: (1) **Expectation-based conceptual analyzers (CAs)**, such as [Dyer,1983], associate each word with one or more knowledge structures, which have rules attached indicating how they can be connected to other structures. This approach has been successful for parsing large pieces of connected text. However, the processing mechanism is overly complex, since each type of knowledge structure generally requires its own set of rules. Parsing highly ambiguous sentences such as S1 and S2 above is particularly problematic since sophisticated back-up and recovery rules are needed. (2) **PDP/Connectionist systems**, examples of which include [Waltz and Pollack,1985], [Cottrell and Small,1985], [McClelland and Kawamoto,1986], have emerged as an alternative to such rule-based approaches. These systems use only simple rules for spreading and combining activation (and in some cases inhibition). Since they are highly parallel and employ scalar activations, complicated backtracking rules are not needed. Unfortunately, these models currently lack operations which are fundamental in higher level NLP systems, specifically: variables, role bindings, instantiations, and inheritance. (3) **Marker passing systems** [Charniak,1986], [Granger et. al, 1986] and [Norvig, 1987], which find connections between concepts by propagating markers over a local semantic network, are a similar approach which provides these high-level operations. Such systems however, generate too many inappropriate connections and typically employ a filter mechanism with its own set of inference rules to weed them out. The complexity of this mechanism negates the simplicity that is the

*This research is supported in part under a contract to the second two authors by the JTF program of the DoD, monitored by JPL and by an ITA Foundation grant to the second author.

advantage of the marker passing approach. (4) **Definite Clause Grammars (DCGs)**, such as [McCord, 1982], unlike the above approaches, focus primarily on the syntactic and structural features of natural language texts, such as conjuncts, quantifiers and agreement. These systems view parsing as a two step process which first constructs a syntactic parse tree through unification and then performs semantic processing. The strength of these systems is their ability to analyze complex linguistic constructs. However, they lack the conceptual information necessary to analyze texts at deeper conceptual levels.

This paper presents CAIN (Conceptual Analyzer for multiple re-INterpretations), which parses highly ambiguous texts while avoiding the problems of the above approaches. CAIN overcomes these problems by: (1) representing all knowledge (both conceptual and structural) in a uniform manner in a local semantic network, (2) using constrained marker passing for all parsing processes, and (3) using link weights, activation values, and thresholds from connectionist models for indicating relative strengths of activations between concepts. Representing all knowledge at the symbolic level provides higher level symbolic operations and allows all parsing processes to be integrated. The marker passing mechanism depends only upon knowledge of the different link and marker types used, so the processing mechanism is simple and independent of the content of memory. Also, since only certain types of marker intersections are considered important and since elements of connectionist models are employed, the problem of spurious connections is avoided. CAIN is implemented in T [Slade, 1987], a Scheme-based dialect of Lisp, and can parse sentences S1 and S2 above.

2. Parsing Using Constrained Marker Passing

The parsing process can be divided into 4 steps: (1) from the input, mark the lexical items and their associated conceptualizations, (2) find the knowledge structures which connect the marked nodes together, (3) bind the roles of these structures, and (4) refine them to be as specific as possible.

The following sections describe how memory is organized and how the above processes are realized using a constrained marker passing and activation mechanism.

2.1 Memory Organization

All knowledge in the system, whether conceptual or structural, is represented using a semantic network, such as that shown in figure 1* below:

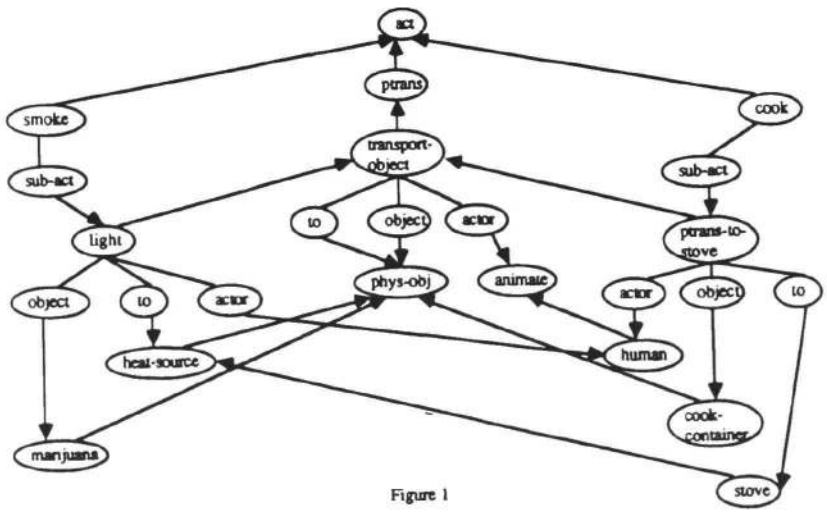


Figure 1

* Due to space limitations, the figures in this paper have been simplified and only show the small portion of the network which is activated by parsing S1.

This figure shows the representation for putting a cooking container on a stove and for lighting a marijuana cigarette. Is-a links, which connect a node to its parents, are represented by the arrows in the figure, and has-a links, which connect a node to its roles, are represented by straight lines. For example, to indicate that lighting a marijuana cigarette is a sub-act of smoking, the node for SMOKE is connected by a has-a link to the node for sub-act, and by an is-a link to the node representing the lighting action. Note that the components of a single act are represented in the same manner. To indicate that the object of the transport action is a physical object, TRANSPORT-OBJECT is connected by a has-a link to its object role, which is in turn connected by an is-a link to PHYS-OBJ. Structural information is represented in the same manner and using the same link types, as illustrated in Figure 2*, which shows the representation for the phrase <person put OBJ1 on OBJ2>, used in parsing S1:

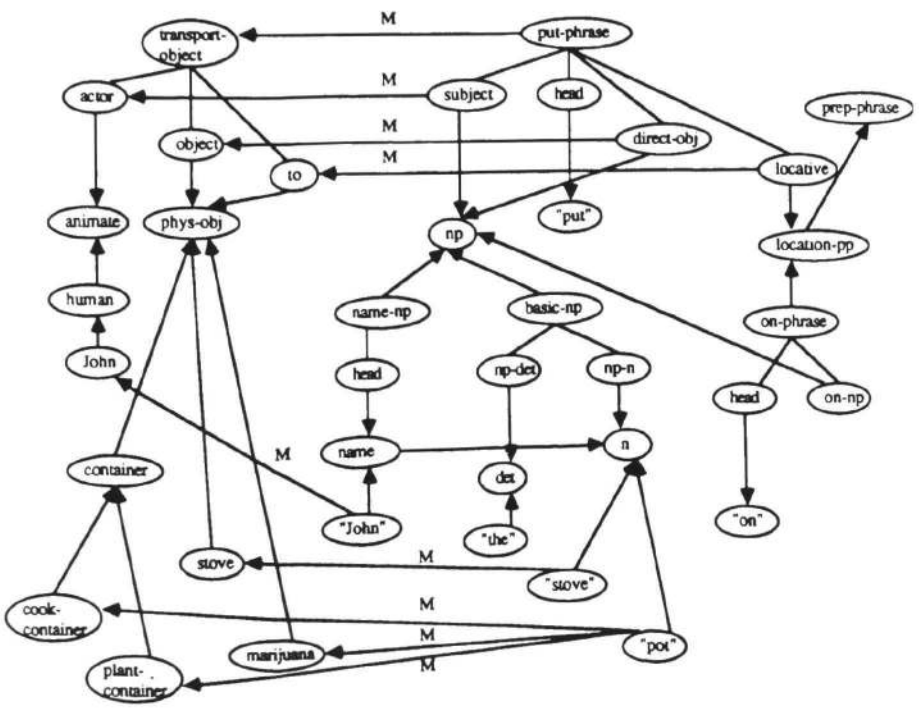


Figure 2

The arrows labeled M (for meaning) in the figure indicate the link between structural and conceptual information.

2.2 Marking the Input Concepts

As the input is read, the occurence of each word and its conceptualizations is indicated by placing an *activation marker (AM)* on the appropriate node. For example, in figure 2, reading the word "pot" results in the placement of AMs on the lexical node "pot" and on the nodes representing the concepts cook-container, plant-container and marijuana. Marking the occurence of a concept also results in the marking of its ancestors, to indicate their implicit occurence. The AM which is placed on COOK-CONTAINER, for example, is also placed on the nodes for container and phys-obj. The rules for marking concepts from the input are therefore:

R-1: When a word is read, an AM is placed on its corresponding lexical node.
R-2: When a lexical or phrasal node receives an AM *of sufficient strength*, an AM is passed across an M link to its associated conceptualizations.
R-3: When a node receives an AM, an AM is passed to its parents.

* This representation is based primarily upon [Gasser, 1988] and [Jacobs, 1985].

AMs which are passed to ancestor nodes contain information indicating the descendant that was the source of the marker. In addition, AMs from lexical nodes also maintain information indicating their meaning(s). This information will later be used to perform role bindings. The reason for the strength constraint on rule R-2 will become clear in subsequent sections.

2.3 Connecting the Input Concepts

How can the correct knowledge structures, connecting the input concepts, be selected? Each node which was activated (received an AM) from the input suggests potentially relevant structures based on the various roles that it plays. This is true for both syntactic and semantic information. For example, since a stove plays the role of an instrument in the cooking schema, activating STOVE suggests that COOK may be applicable. Similiarly, activating the node for determiner indicates that BASIC-NP may be appropriate. *Search Markers (SMs)* are used to indicate knowledge structures which are suggested in this fashion. SMs are propagated according to the following rule:

> R-4: SMs are passed from an activated node, down is-a links to all of its descendants that are role nodes, and across has-a links to the owners of the roles.

Applying the above rule will result in the marking of the correct knowledge structures. However, a large number of inappropriate structures will also receive SMs. For example, when the lexical node for "put" is activated in sentence S1, the above rule will mark the nodes for other phrases involving "put", such as <person-put-up-with-person> and <person-put-on-clothing>, in addition to marking the node for the desired put-phrase shown in figure 2. The solution to this problem is to utilize elements of connectionist models, specifically link weights, activation values and thresholds. Each SM is assigned a *strength value* which depends upon the *weights of the links* over which it is propagated. In general, nodes representing more specific concepts will pass stronger SMs than their ancestors. Thus, the SM that COOK-CONTAINER passes to PTRANS-TO-STOVE in figure 1 will be much stronger than the SM that PHYS-OBJ passes to TRANSPORT-OBJECT. When an SM is propagated to a node representing a knowledge structure, its strength value is added to that of the other SMs on the node. If their combined strengths exceed the node's *threshold level*, then there is strong evidence that the structure is applicable, and it therefore attempts to bind its roles. *Using activation values and thresholds allows a large number of structures to be suggested, while only a few are actively pursued.*

2.4 Role Binding

Binding a role of a structure involves determining whether its filler is activated. If so, then the concept which activated the filler is bound to the role. To bind the subject role of PUT-PHRASE in figure 2, for example, the NP node is checked to determine whether it was previously activated. If it was, the descendant which activated it is then bound to the subject role. The check for whether the filler has an AM is made using a *Role marker (RM)*, which is propagated according to the following rule:

> R-5: When a node's threshold is exceeded, RMs are passed across has-a links to each of its roles, and up is-a links to the fillers of those roles

Note that RMs may be used to indicate roles which should already have been filled or which are expected to become filled. In the latter sense, RMs are very similar to the prediction marker used in DMAP [Riesbeck and Martin, 1986]. Role binding is performed by the rule:

> R-6: When an AM and an RM of sufficient strength intersect*, an AM is placed on the role node

* If it is possible to bind more than one role of a structure to a single concept, then sequencing information, indicating the order in which the roles normally occur, is used to determine which binding is appropriate.

Since an AM maintains information indicating the descendant that was its source, merely placing it on the role has the effect of binding it.

Our confidence in a structure's relevance to the input increases as its roles are bound. For example, as sentence S1 is read and each component of the put-phrase in figure 2 is recognized, it becomes apparent that it correctly represents the input. Thus, binding the subject role should activate the put-phrase node, binding the head role increases its activation level, and similarly for the remaining roles. The amount of the increase depends upon how important the role is to the structure. Role importance is reflected in the strength of the connection between the structure and its roles and therefore in the strength of the RM which is passed by rule R-5. The rule for activating a structure is:

R-7: When an RM and an AM intersect, activate the source of the RM
by placing on it a new AM if one (representing this instance) is not already present,
or by increasing the activation level of the AM which is already there

Syntactic information can be used to help bind roles in semantic structures using rule R-2. For example, when PUT-PHRASE is sufficiently activated by rule R-7, the actor role of TRANSPORT-OBJECT receives an AM (which has John as its meaning) from SUBJECT. Figure 3 shows the AMs which are placed on the structures shown in figure 2, as a result of reading S1.

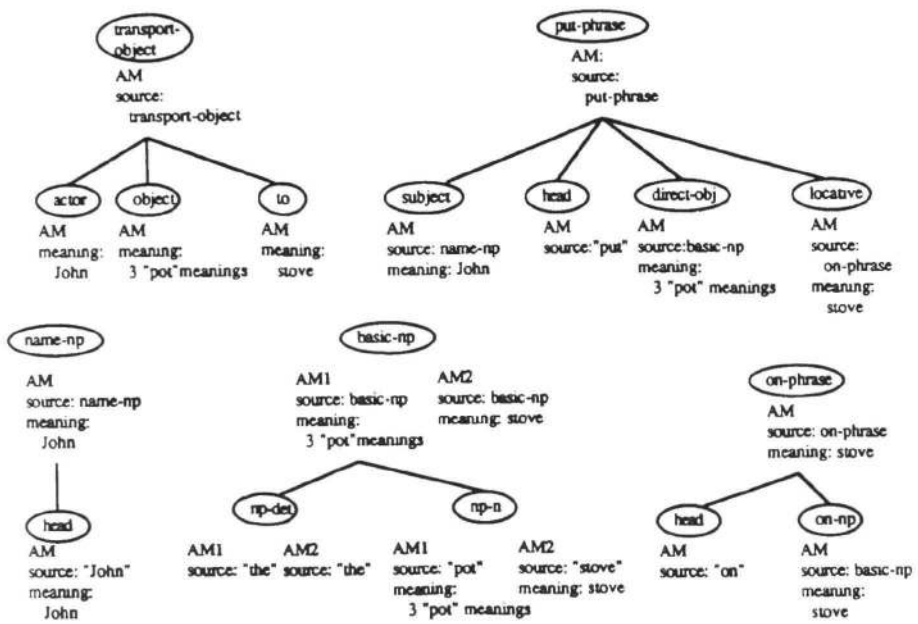


Figure 3

2.5 Concept Refinement

The *most specific* structures possible must be found in order for the input to be completely parsed. A node which is activated by rules R-2 or R-7 may need to be refined to a more specific one using contextual information supplied from the input. Refinement involves searching for a descendant whose equivalent roles have more specific, activated fillers. For example, when the put-phrase is recognized and TRANSPORT-OBJECT is activated by R-2, it can be refined to PTRANS-TO-STOVE as shown in figure 1. The search process is performed using a *descendant marker (DM)* which is spread by the rule:

DM-1: When a node is sufficiently activated by rules R-2 or R-7, a DM is passed down
is-a links to each of its descendants

How is a descendant with more specific fillers found? Recall that each concept which was activated from the input supplies contextual information in the form of SMs, whose strengths are combined when they intersect. A descendant whose fillers are more specific will have SMs with a stronger combined strength. For example, PTRANS-TO-STOVE in figure 1 will receive SMs from HUMAN, COOK-CONTAINER, and STOVE while TRANSPORT-OBJECT will receive SMs from ANIMATE, PHYS-OBJ, and PHYS-OBJ. Since the connection from the roles of PTRANS-TO-STOVE will be much stronger than for TRANSPORT-OBJECT, the former will have a much greater SM strength. Thus, TRANSPORT-OBJECT should be refined to PTRANS-TO-STOVE. The rule which implements concept refinement, then, is:

> DM-2: When a DM is placed on a node whose combined SM level is greater than that of the source of the DM, then bind its roles using the procedure described in section 2.4

Note that after reading S1, TRANSPORT-OBJECT in figure 1 can be refined to either PTRANS-TO-STOVE or LIGHT, so both will be activated. However, since the concept stove suggests cooking much more strongly than HEAT-SOURCE suggests lighting a marijuana cigarette, PTRANS-TO-STOVE will be much more strongly activated. Therefore, it represents the result of the parse. When sentence S2 is read, however, it is recognized as another sub-act of SMOKE. SMOKE will therefore be more strongly activated than COOK, since it receives SMs from two of its sub-act roles, while COOK is unrelated to S2. This causes S1 to be reinterpreted as lighting a marijuana cigarette.

2.6 Marker Removal

As with connectionist systems, markers are removed using a decay process. DMs decay very quickly since they do not have to wait for other nodes to become activated and therefore do not need to remain between sentences. This is not true for the other types of markers, so they decay much more slowly.

3. Related Work

The work presented here was inspired by direct memory access parsing, particularly DMAP [Riesbeck and Martin, 1986]. DMAP attempts to find the most specific knowledge structures that connect the input concepts, using a marker passing algorithm based on recognizing concept sequences. Despite the similarity between DMAP's markers and ours, there are major operational differences. The biggest difference is that DMAP is only able to recognize structures whose roles are encountered in the correct sequence, beginning with the first item. While this works well for syntactic structures which are typically encountered in their entirety and in the correct order, it is not well suited to recognizing higher level conceptual structures such as MOPs [Schank, 1982]. For example, DMAP would not be able to recognize that the COOK context is appropriate after parsing sentence S1, since the intial act, PTRANS-FOOD-TO-CONTAINER, was not encountered. Our work also extends direct memory access parsing (1) to handle ambiguities, reinterpretations, and role bindings, (2) to include more information about syntax, and (3) to represent relative strengths of activations between concepts.

We believe that learning (i.e. adding new nodes and links to the network) will be facilitated by the simplicity of our memory representation. This contrasts to approaches which simplify the processing mechanism by adding extra link types to the network (for example, DMAP uses a special concept refinement link). Our approach is to use only those link types which are necessary for the representation itself and add extra markers where necessary. Since markers are dynamically created during processing and decay with time, adding new ones has no effect on the complexity of the learning mechanism. Similarly, link weights in our model *only* represent relative strengths of connections between concepts and (unlike connectionist systems) are *not* used to control processing.

This work also bears some similarity to SCISOR [Rau, 1987], a system for conceptual information retrieval. The process presented in section 2.3 (for finding the correct structures in memory connected to input concepts) is simi-

lar to the priming rules used in SCISOR. However, SCISOR only addresses the problem of finding episodes in memory, and uses a separate module for parsing. In our work, *parsing and memory search are completely integrated*. Thus, the memory search process described here is more general since SMs can be used to retrieve different types of knowledge structures (such as syntactic information) in addition to retrieving episodes.

4. Conclusions

In this paper, we have presented an approach to parsing natural language texts which integrates a constrained marker passing mechanism with properties of connectionist systems: link weights, activation values and thresholds. This approach is particularly attractive for three reasons. First, it is capable of parsing texts which have proved to be difficult for previous parsing systems, specifically those which are highly ambiguous and require the reader to correct an initially mistaken interpretation. Second, it uses a simple processing mechanism whose rules are independent of the actual content of memory. Thus, new knowledge structures can be added without changing the processing mechanism. Finally, it completely integrates all parsing processes, such as memory search, disambiguation and inferencing.

References

Charniak, E., A Neat Theory of Marker Passing. *Proceedings of the National Conference on Artificial Intelligence (AAAI-86)*, Philadelphia, PA, 1986.

Cottrell, G. and Small S. A connectionist scheme for modeling word-sense disambiguation. *Cognition and Brain Theory, 1*, 89-120, 1985.

Dyer, M.G. *In Depth Understanding*. Cambridge, MA: The MIT Press, 1983.

Gasser, M., *A Connectionist Model of Sentence Generation in a First and Second Language*. UCLA PhD. Thesis, forthcoming.

Granger, R.H., Eiselt, K.P. and Holbrook, J.K., Parsing with Parallelism: A Spreading Activation Model of Inference Processing During Text Understanding. In J. Kolodner and C. Riesbeck (Eds.), *Experience, Memory, and Reasoning*. Hillsdale, NJ: Lawrence Erlbaum, 1986.

Jacobs, P. *A Knowledge-Based Approach to Language Production*. UC Berkeley PhD. Thesis, 1985. Computer Science Division Report UCB/CSD86/254.

McCord, M.C. Using slots and modifiers in logic grammars for natural language, *Artificial Intelligence*, 18 (1982), 327-367.

McClelland, J.L., Kawamoto, A.H. Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences. In McClelland & Rumelhart (eds.) *Parallel Distributed Processing:*. Vol 2. Cambridge, MA: The MIT Press, 1986.

Norvig, P., Inference in Text Understanding, *Proceedings of the National Conference on Artificial Intelligence (AAAI-87)*, Seattle, WA, 1987.

Rau, L.F., Spontaneous Retrieval in a Conceptual Information System. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, Seattle, WA, 1987.

Riesbeck, C.K. and Martin, C.E. Direct Memory Access Parsing. In J. Kolodner and C. Riesbeck (Eds.), *Experience, Memory, and Reasoning*. Hillsdale, NJ: Lawrence Erlbaum, 1986.

Schank, R.C. *Dynamic Memory: A theory of Reminding and Learning in Computers and People*. NY: Cambridge University Press, 1982.

Slade, S. *The T Programming Language*, Englewood Cliffs, NJ: Prentice-Hall, 1987.

Waltz, D. and Pollack, J. Massively Parallel Parsing: A strongly interactive model of natural language interpretation, *Cognitive Science*, 9 (1985), 51-74.

Learning Subgoals and Methods for Solving Problems

Richard Catrambone
The University of Michigan
Department of Psychology

Keith J. Holyoak
The University of California at Los Angeles
Department of Psychology

In prior studies (Catrambone & Holyoak, 1987) we demonstrated that subjects studying examples dealing with the Poisson distribution tended to learn a series of steps for solving problems rather than learning meaningful chunks. We argued that these chunks should consist of relevant subgoals and methods for achieving those subgoals. If a person learns subgoals and methods for a particular problem domain, then they should be better able to solve novel problems in that domain. Novel problems are defined as 1) having a subgoal order that is different from previously studied problems and/or 2) requiring modifications of old methods for achieving one or more subgoals.

One encouraging finding from our earlier experiments, from a pedagogic point of view, was that if subjects studied examples that demonstrated two different solution procedures, then they were more likely to learn some of the subgoals in the problem space. However, subjects still had difficulty modifying old methods. Two limitations with the earlier experiments were that subjects only studied four examples and the procedures differed from each other in several ways. The first limitation made transfer more difficult to achieve and the second limitation made comparisons among groups less straightforward that we would have liked. The purpose of the present paper is to describe an experiment that gave subjects more extensive training and carefully manipulated just the methods subjects learned for finding subgoals. We wished to examine whether more extensive training with two or more methods for finding a subgoal would help learners to figure out how to modify those methods in new problems.

Our work has suggested that people's procedural knowledge for a domain such as solving probability problems can be represented in terms of subgoals and methods. These subgoals and methods may be based on the "true" subgoals and methods in the domain, or they may correspond to superficial features of example problems. There may be no agreement over what are the "true" structural features for a problem domain, but there is certainly more agreement among experts than novices in a domain about the domain's relevant features (e.g., Adelson, 1981; McKeithen & Reitman, 1981).

Related Work

Various researchers have shown that learners focus on superficial features of examples and need to be guided to focus on the deeper aspects. Anderson and his colleagues (Anderson, Farrell, & Sauers, 1984; Pirolli & Anderson, 1985) demonstrated that students learning how to write recursive functions in the LISP programming language relied heavily on the syntactic elements of examples in order to create their first few functions. These students essentially mapped each part of the current problem onto an example. This approach worked because the early target problems were designed to be isomorphic with the examples. However, students must eventually be weaned from heavy reliance on pure example mapping and induced to focus on subgoals and methods in order to be able to solve more difficult or novel problems in a domain.

McKendree (1986) found that geometry students performed best on new problems when they received error feedback on examples that focused on subgoals rather than feedback that simply told the student that he or she had made an error. In addition, an earlier protocol study of tutors

indicated that a very important role of feedback was to make the novice's goals explicit and to help keep track of these goals during the task (McKendree, Reiser, & Anderson, 1984).

Charney and Reder (1986) showed that diverse examples are important in helping people learn procedures and when to apply them (i.e., to learn selection rules). Their subjects studied examples in order to learn various commands on a personal computer. Charney and Reder found that varied examples, which demonstrated a range of situations in which a command could be used, helped new users grasp the utility and applicability of the command.

The studies cited above suggest that people learning information in a new domain need to be directed to focus on the relevant aspects of training materials. Examples must be carefully chosen to help the user gain a functional understanding of the subgoals, methods, and when to apply the methods (Charney & Reder, 1986; VanLehn, 1982, 1985). In addition, learners must be able to modify methods in new situations. This last issue was an important focus in the current experiment which dealt with subjects learning how to solve problems dealing with the Poisson distribution.

The Poisson Distribution and Some Examples

The Poisson distribution is often used to approximate binomial probabilities for events that occur with some small probability p. The Poisson equation is: $P(X=x)=[e^{-\lambda}\lambda^{x}]/x!$, where λ is the expected value--the average--of the random variable X. One use of the Poisson distribution is illustrated in Table 1.

The subgoals and methods (in parentheses) for this "quarry" problem could be listed as follows:

1) find λ (find total frequency of event and divide by total number of trials)
2) find expected probability for each X (plug each X into the Poisson equation)
3) find expected frequency for each X (multiply each P(X) by the total number of trials).

In prior experiments (Catrambone & Holyoak, 1987) we found that subjects could study a problem such as the one in Table 1 and then solve an isomorphic problem such as one dealing with the number of errors made per game by a baseball team's infielders. However, these subjects had great difficulty solving the "birthday" problem in Table 2 that had a different subgoal order and required a modified method to find λ.

In the birthday problem the solver must realize that λ is simply the frequency of the event (number of people in the room) divided by the number of trials (days of the year). Once λ is found, then it can be put into the Poisson equation along with the value of X that is desired in order to find the expected probability for that value of X. Most subjects did not realize that λ was a subgoal that could be found using a modification of the method used in the "quarry" problem (where the event frequency was not given directly). In addition, few subjects even realized they could find the expected probability for a particular value of X. The protocols of most of the subjects indicated that all they really knew how to do was apply a series of steps they had learned from the "quarry" problem. They typically did not differentiate the steps into methods for reaching particular subgoals.

Table 1
"Quarry" Problem Using the Poisson Distribution

A horizontal quarry surface was divided into 30 squares about 1 meter on a side. In each square the number of specimens of the extinct mammal *Ditolestes motissimus* was counted. The results are given in the table below. Fit a Poisson distribution to x, that is, give the expected frequencies for the different values of x based on the Poisson model.

Number of Specimens per Square	Observed Frequency
0	16
1	9
2	3
3	1
4 or more	1
Total	30

SOLUTION:

$E(X) = [0(16)+1(9)+2(3)+3(1)+4(1)]/30 = 22/30 = .733 = \lambda$
=average number of specimens found in each square

$P(X=x) = [(e^{-.733})(.733^x)]/x! = [(.48)(.733)^x]/x!$

Fitted Poisson Distribution:

x	Expected Frequency
0	.48 * 30 = 14
1	.352 * 30 = 11
2	.1289 * 30 = 4
3	.0315 * 30 = 1
4 or more	.0058 * 30 = 0

Table 2
"Birthday" Problem Using the Poisson Distribution

Suppose you took a random sample of 500 people and found out their birthdays. A "success" is recorded each time a person's birthday turns out to be January 1st. Assume there are 365 days in a year, each equally likely to be a randomly chosen person's birthday. Fit a Poisson distribution to x (the number of people born on January 1st) and find the predicted likelihood that exactly 3 people from the sample are born on January 1st.

SOLUTION:

$\lambda = 500/365 = 1.37$ = average number of people born on any given day

$P(X=3) = [(e^{-1.37})(1.37^3)]/3!$
$=[(.254)(2.57)]/6$
$=.109$
=likelihood of exactly three people being born on January 1st

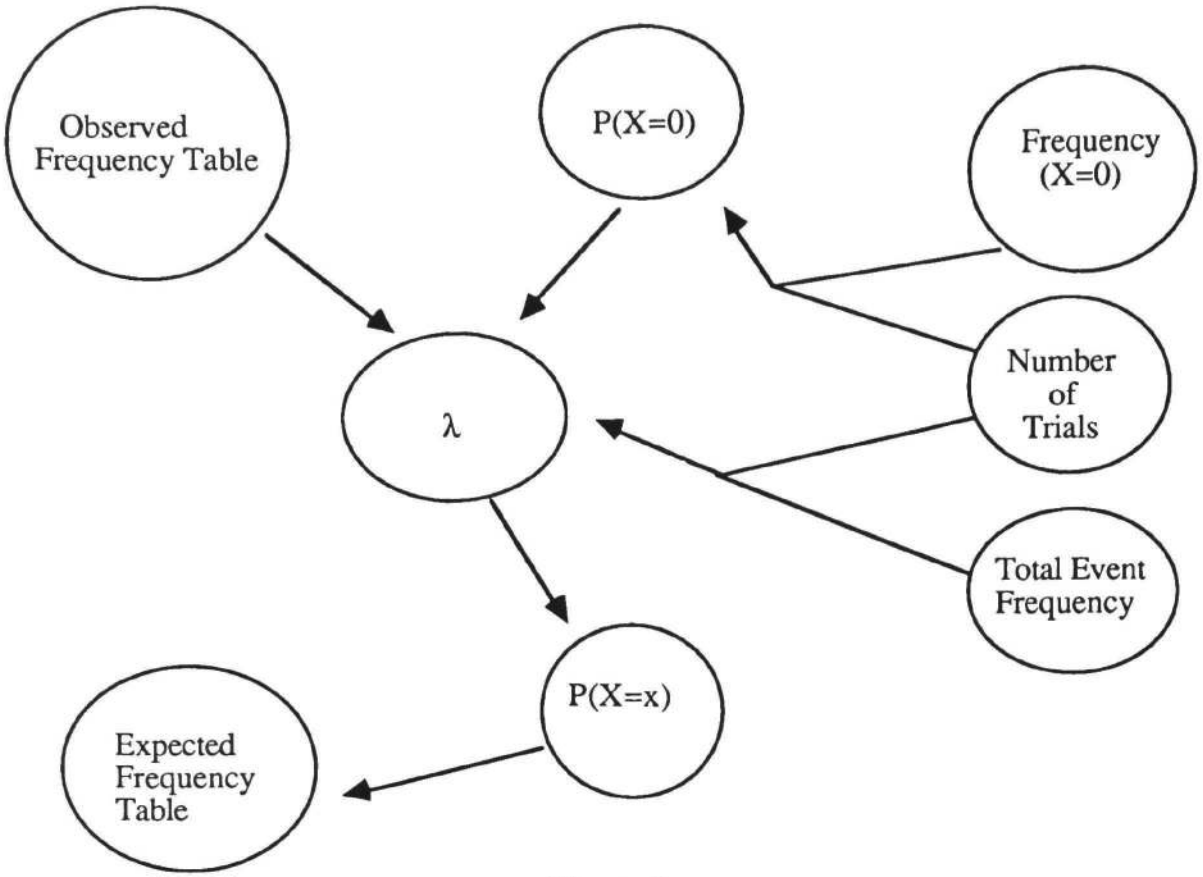


Figure 1
Problem Space for Poisson Distribution Problems

Figure 1 shows the problem space for Poisson distribution problems. The circles represent possible subgoals (or givens) and the arrows indicate which subgoals can be reached from other subgoals. The figure shows that λ can be reached in a few ways.

In our earlier experiments (Catrambone & Holyoak, 1987) we found that subjects who studied problems only of the quarry type recognized the subgoal of finding λ in the birthday problem about 30% of the time and the subgoal of finding P(X=x) about 35% of the time. In addition, we found that subjects who studied problems of the quarry type and another type (in which P(X=0) was given and the problem asked for the value of λ) recognized the subgoal of finding λ about 50% of the time and finding P(X=x) 98% of the time. This suggests that studying multiple solution procedures helped make subgoals (such as finding λ) more apparent to learners. However, one thing these subjects could not do very well was figure out <u>how</u> to find λ. Half of the subjects knew that λ needed to be found but did not know how to get it other than to use the frequency table method demonstrated in the quarry problem. They were not able to modify this method.

Current Experiment

In the current experiment , we solicited paid volunteers from an upper-level probability course at the University of Michigan. These students had been taught about random variables and were in the process of learning the binomial distribution. Students participated in the present experiment,

which dealt with the Poisson distribution, while learning the binomial distribution but before learning the Poisson. We sought to manipulate more systematically the methods for finding λ that subjects studied and to increase the number of examples of each method. An extensive discussion of the design of the experiment is in Catrambone and Holyoak (in preparation). Below is an outline of the relevant aspects of the design.

Procedure

Subjects studied 10 examples which demonstrated different methods for finding λ. Methods for finding λ are listed below. Some subjects learned two methods and other subjects learned three methods. No subject studied methods #4 or #5. In the test phase, subjects solved 10 problems using all five λ-methods.

The five methods for finding λ are:

1) λ given directly in the problem
2) Calculated from an observed frequency table
3) Calculated from the Poisson equation if P(X=0) is given
4) Calculated from the Poisson equation if the frequency of X=0 is given and the total number of trials is given
5) Calculated by dividing the frequency of the event by the total number of trials

Results and Discussion

Figure 2 presents the percentage of subjects that used λ-methods #4 and #5 correctly in the test phase as a function of the number of λ-methods they studied in the training phase. Subjects in both groups were quite good using λ-method #4. This result is not too surprising since λ-method #4 is quite similar to λ-method #3 and all subjects studied λ-method #3. The only difference in the methods is that #4 requires the solver to find P(X=0) rather than being given it directly. The prior studies had indicated that subjects were quite good at going from a frequency to a probability. A more interesting result is that subjects used λ-method #5 correctly about 75% of the time. This method is a simpler version of λ-method #2. In #2 the solver must find the frequency of the event whereas in #5 the frequency is given directly. In the prior studies, subjects were typically quite unsuccessful in adapting λ-method #2 into #5. The highest success rate of any of the earlier groups was 53%. The performance of subjects in the current experiment was significantly better than that (75% vs 53%, $z = 2.44$, $p < .02$).

This would suggest that the more extensive practice with example problems benefitted subjects in the current experiment. The manipulation of two versus three types of methods did not seem to make a great deal of difference. It would seem that extensive practice with at least two methods is the critical feature for being able to modify the old methods. The particular methods that are learned also matters (see Catrambone & Holyoak, in preparation, for discussion of this issue).

The prior experiments combined with the current one suggest that subjects who studied example problems that illustrated multiple methods for finding a particular subgoal were able to recognize that subgoal more successfully than subjects who only studied one procedure. There are several explanations for this result. One explanation is related to the work of Sweller and his colleagues (Mawer & Sweller, 1982; Sweller & Levine, 1982; Sweller, Mawer, & Ward, 1983) that indicates that when problem solvers are led to focus on a single ultimate goal, they concentrate on reducing the distance between the initial state and the goal state and thus fail to recognize subgoals and

methods in the domain. The one-procedure subjects in the earlier experiments learned a single concrete goal when they studied the example problems such as taking a frequency table as input and, through a stereotyped series of operations, producing an expected frequency table as output. They did not form intermediate goals such as finding λ or a single P(X). The multiple-procedure subjects, by virtue of seeing the same unknown (such as λ) reached in more than one way, were able to recognize that unknown as a subgoal. The extensive practice on two or more methods by subjects in the current experiment helped them to modify old methods better than subjects in the earlier studies.

Extensive practice with multiple methods appears to be an important component in enabling learners to recognize subgoals or modify methods, at least to some degree. That is, learners are reasonably successful in altering methods if they have had a fair amount of practice with two or more. It will be important to continue to examine the relative contributions of variety and amount of practice on learners' ability to isolate subgoals and methods and to modify the methods for a variety of domains. In addition, it will be important to study <u>how</u> practice allows people to modify methods.

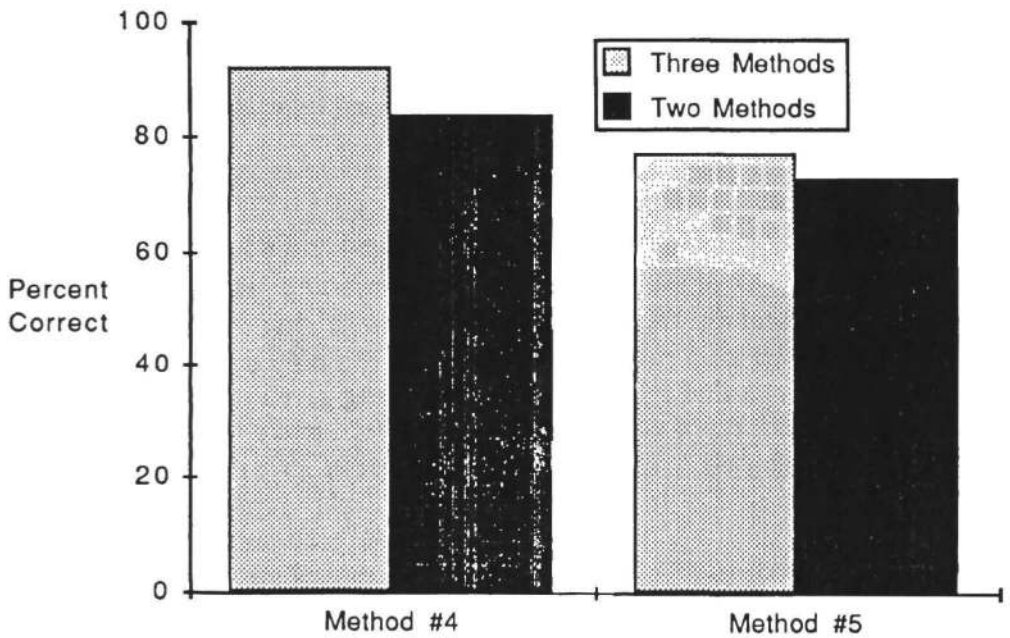


Figure 2
Success at Using New λ-Methods as a Function of the Number of Methods Studied During Training

References

Adelson, B. (1981). Problem solving and the development of abstract categories in programming languages. Memory & Cognition, 9, 422-433.

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

Catrambone, R., & Holyoak, K.J. (1987). Transfer in problem solving as a function of the procedural variety of training examples. In Proceedings of the Ninth Annual Conference of the Cognitive Science Society. Hillsdale, NJ: Erlbaum, 36-49.

Catrambone, R., & Holyoak, K.J. (in preparation). The impact of procedural variety of training examples on problem-solving transfer.

Charney, D.H., & Reder, L.M. (1986). Initial skill learning: An analysis of how elaborations facilitate the three components (Tech. Rep. No. ONR-86-1). Pittsburgh: Carnegie-Mellon University.

Mawer, R., & Sweller, J. (1982). The effects of subgoal density and location on learning during problem solving. Journal of Experimental Psychology: Learning, Memory, and Cognition, 8, 252-259.

McKeithen, K.B., & Reitman, J.S. (1981). Knowledge organization and skill differences in computer programmers. Cognitive Psychology, 13, 307-325.

McKendree, J. (1986). Impact of feedback content during complex skill acquisition. Ph.D. Dissertation, Carnegie-Mellon University.

McKendree, J., Reiser, B.J., & Anderson, J.R. (1984). Tutor goals and strategies in the instruction of programming skills. In Proceedings of the Sixth Annual Conference of the Cognitive Science Society, Boulder, CO, 252-254.

Pirolli, P.L., & Anderson, J.R. (1985). The role of learning from examples in the acquisition of recursive programming skill. Canadian Journal of Psychology, 39, 240-272.

Sweller, J., & Levine, M. (1982). Effects of goal specificity on means-ends analysis and learning. Journal of Experimental Psychology: Learning, Memory, and Cognition, 8, 463-474.

Sweller, J., Mawer, R.F., & Ward, M.R. (1983). Development of expertise in mathematical problem solving. Journal of Experimental Psychology: General, 112, 639-661.

VanLehn, K. (1982). Bugs are not enough: Empirical studies of bugs, impasses, and repairs in procedural skills. The Journal of Mathematical Behavior, 3, 3-71.

VanLehn, K. (1985). Arithmetic procedures are induced from examples, (Tech. Report No. ISL-12). Xerox Palo Alto Research Center.

Opportunistic Use of Schemata for Medical Diagnosis*

Roy M. Turner
School of ICS
Georgia Institute of Technology

Medical diagnosis can be considered a planning task. This is not the traditional view, however. For example, Gomez and Chandrasekaran (1982) and others view diagnosis as a classification task: a problem, consisting of a set of signs and symptoms, is classified as being an instance of a disease or set of diseases. However, this viewpoint overlooks the fact that actions are performed in order to classify a disease: in other words, planning and plan execution must be done as part of the classification process. When viewed as planning, goals in diagnosis are such things as "diagnose the patient," "interpret a finding," and "evaluate a hypothesis." Operators, at the lowest level, are such things as asking questions, requesting tests be performed, and making inferences based on information known about the patient and the reasoner's general knowledge of the domain.

Medical diagnosis is unlike many traditional planning tasks in that an initial, complete statement of the problem is generally impossible. Instead, the diagnostician must gather information about the problem as part of the process of performing diagnosis. The result of this is that the diagnostician cannot formulate a plan for diagnosis, then carry it out: the problem statement would change as the plan for performing diagnosis is executed. The effect of executing one step (e.g., asking a question) would likely alter the assumptions upon which later steps are based (e.g., a new finding might radically alter the diseases considered as diagnoses, or might suggest specialized methods for interpreting the finding). The problem for a diagnostician, then, is to be able to interleave planning and execution (cf. McDermott, 1978) so as to make use of new information as it becomes available. In other words, a diagnostician should be *opportunistic*.

*This research has been funded in part by NSF Grants IST-831771 and IST-8608362 and grant DTD 09-25-87 from the Lockheed AI Center.

Our approach to this problem makes use of packets of procedural information called *schemata*, which are retrieved from memory in response to goals arising from changes in the problem solver's environment: e.g., new findings, new hypotheses, etc. Most schemata can achieve very specific goals, such as "interpret a finding" or "evaluate a hypothesis"; others control larger parts of the reasoner's processing, such as directing the reasoner in the overall consultation. Schemata are flexible enough to encode several variations of how to achieve their goal; in addition, specializations of schemata provide the reasoner with information about how to satisfy specific goals or goals arising in specific contexts.

When a goal arises that can be achieved by a schema, that schema is retrieved from memory and made active. As the reasoner may have many goals simultaneously, there may be many active schemata at any time. The reasoner must decide which goal to focus on, and hence, which schema to *apply*. In our approach, the reasoner uses information from two sources to help it focus its attention. One source is from memory structures representing generalized consultations similar to the current problem. Information from these generalized consultations, such as information about which findings are generally important in this context, can be used by the reasoner to help it select a goal to achieve. The second source is from packets of procedural knowledge, called *strategic schemata*, which contain generally useful strategies in the form of goal orderings: e.g., a medical reasoner would have strategies for performing hypothetico-deductive reasoning, reasoning under time pressure, etc.

In this paper, we discuss our approach to opportunism in medical diagnosis. Our approach is called *schema-based reasoning*. Our ideas are being tested in MEDIC (Turner, in press), a schema-based diagnostic reasoner whose domain is pulmonology.

OPPORTUNISM USING SCHEMATA

Opportunism involves responding to changes in the task environment as they arise during problem solving. There are at least three capabilities a reasoner must have in order for it to respond to changes; it must be able:

1. to interleave planning to achieve a goal and execution of that plan;
2. to respond immediately to new information appearing in the environment; and
3. to select the appropriate goal to pursue at each point in problem solving—that is, it must be able to focus its attention.

Traditional planners do not interleave planning and execution. Instead, the planner formulates a plan, then applies it. On the other hand, rule-based problem solvers and purely reactive planners such as PENGI (Agre & Chapman, 1987) do not really perform planning *per se*. The problem with these approaches is that there is very little coherence in their actions; consequently, their behavior may seem strange and unintuitive to a user. This presents a problem, especially in a medical domain, since a user is unlikely to accept a system if he or she cannot understand its reasoning.

A middle ground is needed between traditional planners and purely reactive planning. In our approach, problem solving is carried out by retrieving packets of procedural knowledge from memory, then applying them. These packets, or *schemata*, can be thought of as small plans or pieces of plans that achieve a goal; for instance, a reasoner may have a schema which can interpret a finding or one that can evaluate the likelihood of a hypothesis that a particular disease is present. Figure 1 shows a simplified view of a schema for interpreting a finding of dyspnea.[1] A schema contains *steps* to be performed by the reasoner in order to satisfy a particular goal.

Our approach is more flexible than traditional planning for three reasons. First, the order of the steps of a schema is not completely fixed ahead of time, but rather depends, to some extent, on the situation at the time of schema execution. For example, the step labeled "S1" contains information that allows the reasoner to select the next step based on the answer to the question asked in S1.

[1]Shortness of breath.

```
Goal: interpret a finding of dyspnea
Patient: any patient
Findings: dyspnea
Preconditions: there is a finding of dyspnea
Steps:
    S1:  action: ask how many stairs patient can climb
         goal: determine severity of dyspnea
         next:
              if pt. can climb flight of stairs ==> S3
              else ==> S2
    S2:  action: ask how far patient can walk
         goal: determine severity of dyspnea
         next: S3
    S3:  action: estimate the severity of the dyspnea
         goal: determine severity of dyspnea
                  :
    S10: action: postulate hypotheses of pulmonary disease,
         cardiac disease
         goal: explain dyspnea
         next: done
Indices:
    patient/PATIENT1 -> SCENE2
          :
```

Figure 1: sc-dyspnea—a schema for interpreting a finding of dyspnea.

The second source of flexibility in our approach is also due to the nature of a schema's steps. In addition to specifying actions that should be taken—either primitive actions or other schemata—a step in a schema usually specifies the goal that the step is to satisfy. If the step fails, the reasoner can attempt to find another way of satisfying the step at run time. In addition, a step does not necessarily specify an action. Instead, it can specify *only* a goal, thus forcing the reasoner to attempt to satisfy the goal at run time.

The third reason our approach is flexible is that a single plan is not formulated for all of the goals in a problem, then executed. Instead, individual schemata are retrieved and applied to satisfy goals. As the situation changes, which schemata are active will also change. For example, as new goals arise during problem solving, new schemata can be found and activated to satisfy them.

In order to exhibit opportunism, a reasoner must be able to notice and respond to new information as it becomes available. In the context of a diagnosis program, new information comes from the user; information may be volunteered, or it may come from answers to questions asked by the system. In either case, when new information becomes available, the reasoner should interrupt what it is doing and incorporate the information into what it knows; the new information may also cause the reasoner to alter the course of its problem-solving behavior.

In our approach, all actions are governed by information contained in schemata; thus, to handle a piece of information, a schema must be found and activated. When a new item of information is encountered, the reasoner interrupts what it is doing and looks for a schema that can satisfy the goal created by the occurrence of the information: e.g., if the information is a finding, the goal will be to interpret it. When a schema is found, it is activated. The reasoner can then either return to what it was doing (i.e., to the schema it was applying), or it can choose, based on the altered state of the environment, to pursue a different goal (i.e., to apply a different schema—perhaps the one just activated).

There may be many goals present for a given diagnostic problem: e.g., goals to interpret findings, evaluate hypotheses, and to produce a diagnosis. Each of these will lead to one or more schemata being activated. In addition, information learned during diagnosis will result in more schemata being activated, as discussed above. Once we allow the reasoner to have more than one schema active at a time, we face the need for ability (3) above: the reasoner must be able to select the appropriate schema to apply at each point in problem solving;[2] i.e., the reasoner must decide on which goal to focus its attention.

This is not an easy task, since the importance of a goal varies with context. For example, in one situation, a goal to explain a severe rash may be quite important; however, in another situation involving both a severe rash and hemoptysis,[3] the goal of explaining the hemoptysis should take precedence over the goal of explaining the rash.

One approach to this problem is to use knowledge about the type of consultation the reasoner is performing. This information, in our approach, is present in memory structures called *diagnostic memory organization packets*, or dxMOPs (cf. Schank, 1982; Kolodner, 1985; and the diagnostic categories of Kolodner and Kolodner, 1987). These structures participate in memory organization, and provide one way for the reasoner to retrieve schemata from memory (see (Turner, in press)

```
Goal: diagnose the patient
Patient: an alcoholic patient
Chief complaint: dyspnea
Findings:
    anemia: low importance, explained by
            alcoholism
    ataxia: low importance, explained by
            alcoholism
        ⋮
Hypotheses: TB, sarcoid, generalized pulmonary or heart
            disease
Schemata:
    sc-consult: for goal of diagnosing patient
    sc-finding: for generic findings
    sc-TB: for evaluating TB
        ⋮
Indices:
    finding/mass-on-X-ray → dxMOP3
        ⋮
    patient/PATIENT4 → dxMOP2
        ⋮
```

Figure 2: A dxMOP for consultations involving alcoholics with dyspnea.

for details). Each dxMOP represents a particular class of consultations, and provides information about goals, actions, etc., that can be expected in such a consultation. Information from a dxMOP can be used to decide which goal—and hence, which schema—to pursue. For example, suppose the dxMOP is the one shown in Figure 2; this dxMOP represents consultations involving alcoholics whose chief complaint is dyspnea. Among the expected findings are anemia and ataxia,[4] which are both explained by alcoholism. If the reasoner discovers that the patient indeed has anemia, it would not need to follow it up, since the finding is anticipated by the dxMOP and marked with a low importance.

Information contained in a dxMOP can help the reasoner order goals; but what if there is no such information present in the dxMOP retrieved from memory? Or what if there are constraints associated with the current problem that are not anticipated in the dxMOP, such as time being limited?

In order to focus the reasoner's attention in this type of situation, we use the idea of meta-reasoning (e.g., Davis & Buchanan, 1984): reasoning that takes place to guide planning. In our approach, meta-reasoning information is present in the form of strategic knowledge structures called *strategic schemata*. A strategic schema is a packet of information which represents a strategy for the rea-

[2] We do not allow schemata to be applied in parallel. This is because we are trying, as far as possible, to model the behavior of human diagnosticians, who generally act as though they are thinking of one thing at a time. There are two reasons for modeling humans: (1) if the reasoner behaves similarly to a human, then the user is more likely to understand its reasoning, and hence, accept it; and (2) reasoning in a manner similar to that of a human should make explanations easier (though we do not currently address explanation).

[3] Blood in the sputum.

[4] Poor motor coordination.

```
Situation: any
Goal ordering:
    select goals related to hypotheses
    select goals related to findings
    select goals for gathering information
    select goal for forming diagnosis
```

Figure 3: A simple strategy for hypothetico-deductive reasoning style.

```
Situation: time is short
Goal ordering:
    select goal related to chief complaint
    select goals related to hypotheses
    select finding goal only if very important
    select goals of forming diagnosis
```

Figure 4: A simple strategy for reasoning under time pressure.

```
begin
    loop forever:
        Wait until user requests a consultation;
        Add goal of diagnosing patient to short-term
                memory;
        Retrieve dxMOP using goal;
        Use strategy from dxMOP, if possible;
        Select a schema from the dxMOP to satisfy
                goal, add it to agenda;
        loop until done:
            Select a schema from agenda using strategies,
                local information in the dxMOP;
            Apply one action;
            if there was an interruption then:
                Handle interruption;
            fi;
            Specialize current dxMOP;
            if specialization succeeded then:
                Set current dxMOP to be the
                        specialization;
            fi;
        end loop;
        Accept and process feedback;
        Update memory;
    end loop;
end.
```

Figure 5: Basic schema-based reasoning algorithm.

soner's behavior; since the reasoner's behavior is determined by which schemata it chooses to apply, a strategy is equivalent to an ordering of the schemata applied. Strategic schemata are useful for representing general strategies such as hypothetico-deductive reasoning or reasoning under time pressure. The reasoner can then use information, if available, from a dxMOP to modify its use of these strategies for a specific situation.

A simple example of a strategic schema is shown in Figure 3. This schema provides a goal ordering for the reasoner that induces a crude form of hypothetico-deductive reasoning: select any goals (i.e., select their corresponding schemata) that relate to hypotheses first; if there are none, then select goals related to findings in the hope of producing hypotheses; if none, then select goal of gathering information from the user; and finally, if that cannot be done, select the goal of forming a diagnosis. Figure 4 shows a simple strategic schema for reasoning under time pressure: follow up the chief complaint, if possible; evaluate hypotheses; only select goals related to findings if the findings are very important; and finally, when all else fails, form a diagnosis.

The overall algorithm for schema-based reasoning, including opportunism, is shown in Figure 5. Note that the reasoner can be interrupted; these interruptions can include interruptions both by the user and by schema application, as new information is added to STM.

MEDIC

Our approach to diagnostic reasoning is being implemented in the MEDIC program, a schema-based reasoner which performs diagnosis in the domain of pulmonology. MEDIC consists of three major modules: a long-term memory, which is organized as described in (Turner, in press); a short-term memory (STM); and a schema-based reasoner, which is directed at all times by schemata.

Conceptually, there are three types of schemata in MEDIC's memory: global, local, and strategic schemata. A *global schema* is one that directs a major portion of a consultation; an example is the schema which contains information that the reasoner can use to conduct the consultation: ask for a patient description, ask about the chief complaint, gather information, then form a diagnosis. Gathering information and forming a diagnosis are also directed by global schemata. A *local schema* is one which directs the reasoner in achieving very specific goals: e.g., interpret a finding of dyspnea or evaluate the hypothesis of lung cancer. Strategic schemata represent general reasoning strategies and are described above.

Currently, MEDIC can diagnose very simple cases of pulmonary disease. MEDIC follows an algorithm very similar to that in Figure 5. Let's look at an

```
Please describe the patient.
: (patient (sex female) (weight (value 204)) (height
            (value 64)) (race white))
Adding information about patient to STM.
What is the chief complaint?
: (finding (entity (dyspnea (duration (years 2))
            (character progressive))))
Adding chief complaint to STM...adding finding of
            <DYSPNEA0> to STM.
How many flights of stairs can the patient climb?
: (less-than 1)
How far can the patient walk on level ground?
: (yards 20)
I judge the qualitative value of SEVERITY of <DYSPNEA0>
            to be SEVERE.
            ...(same for cardiac disease)...
...explaining dyspnea...
Processing <HYPOTHESIS0> [pulmonary disease];
            relating to other hypotheses...
...generating expectations given <HYPOTHESIS0>...
...I'm scoring hypothesis <HYPOTHESIS0> (<PULM-DZ0>)
...hypothesis explains: (<FINDING0>) [dyspnea]...
...failed predictions for hypothesis: —
...hypothesis doesn't explain: —
...trying to specialize the hypothesis of
            <HYPOTHESIS0> (<PULM-DZ0>)...
...specialized <HYPOTHESIS0> to <HYPOTHESIS1>
            (<RPE>) [recurrent pulmonary embolism]
...generating expectations given <HYPOTHESIS1>...
            ...
Is there a finding of <SYNCOPE>?
: Yes
            ...
Enter information (<return> if no more).
:
            ...
My diagnosis is: Recurrent pulmonary embolism.
```

Figure 6: Part of a consultation with MEDIC.

example of a consultation with MEDIC, a portion of which is shown in Figure 6. Suppose a user requests a consultation. The reasoner looks in memory for a way of satisfying the goal of diagnosing a patient and finds a dxMOP, "dx-consult", representing how consultations are generally conducted. The reasoner then uses this dxMOP as a context for diagnosis: it is used as a source both of a strategy and of schemata to satisfy active goals. The strategy it contains is "st-HD-reasoning", the strategic schema mentioned above which provides a goal ordering to induce hypothetico-deductive reasoning. The only goal active is one to diagnose the patient; the schema to achieve this in dx-consult is "sc-consult". This is added to the reasoner's agenda of active schemata.

The reasoner now selects a schema from its agenda, using the goal ordering provided by the current strategy; the use of specific information from the dxMOP is not currently implemented. The only schema to select is sc-consult, so the reasoner selects that and begins to apply it. The user is asked for some initial information about the patient, including a description of the patient (a white female who is overweight)[5] and the chief complaint (progressive dyspnea). The information is added to STM. Adding the chief complaint causes the reasoner to be interrupted, and it searches memory for a schema to interpret the finding. Schema "sc-dyspnea" is found and activated.

Note the opportunistic nature of this: a piece of information appears in the reasoner's environment, which causes the reasoner to interrupt what it is doing in order to respond. The response involves finding a method of interpreting the finding and activating it, resulting in more than one active schema. The next step is to choose one to apply. Since the strategy in use dictates that goals related to findings have precedence over goals for gathering information or forming a diagnosis, the new schema, sc-dyspnea, is selected and used. The new information has effectively changed the course of the reasoner's problem-solving behavior.

This schema, sc-dyspnea, is a specialized version of a general schema to interpret findings; instead of asking general questions, the schema can ask very specific things related to dyspnea (e.g., asking how many stairs the patient can climb as a measure of the severity). The last step of this schema is to explain the finding by postulating diseases that could cause it; using this step, the reasoner postulates hypotheses of pulmonary disease and cardiac disease. Adding these hypotheses to STM again interrupts the reasoner, which finds and adds to the agenda schemata to evaluate the hypotheses: "sc-pulmDz" and "sc-cardiacDz".

The strategy orders goals related to hypotheses before any others; hence, one of the two schemata just added is selected, in this case, sc-pulmDz. The reasoner uses this schema to score the hypothesis of pulmonary disease,[6] and then tries to specialize the hypothesis using information that is in STM. One possible specialization, based on the fact that the patient is overweight, is recurrent pulmonary embolism[7] (RPE); this is hypothesized, resulting in a schema ("sc-RPE") being activated to evaluate it.

The reasoner then selects sc-RPE and begins to evaluate the hypothesis of pulmonary embolism. Eventually, it will have evaluated all the hypotheses that it can and will have exhausted the information the user can give it. The main schema, sc-consult,

[5] Input to MEDIC is in a version of Conceptual Dependency (Schank and Abelson, 1977); there is currently no natural language interface.

[6] Using a scoring scheme very similar to that of INTERNIST-1 (Miller *et al.*, 1982).

[7] Blood clots occurring in the lungs.

will then suggest the step of forming a diagnosis, which will be attempted.[8] In this case, the best hypothesis is recurrent pulmonary embolism, and that will be proposed to the user.

There is still much work to be done on MEDIC. At the present, the program has very little domain knowledge, and relatively few schemata. Additional thought must also be given to the form and content of the strategic schemata, which are currently quite simple; eventually, we would like for them to specify actions for the reasoner to perform in order to select a goal to pursue, making them more like the reasoner's other schemata. MEDIC also does not make use of context-specific information in dxMOPs to focus its attention.

RELATED WORK

Opportunistic reasoning has been addressed in the blackboard approach to problem solving of HEARSAY-II (Erman *et al.*, 1980) and OPM (Hayes-Roth, 1985). These program's *Knowledge Sources* (KS's) are atomic, rule-like specialists that are invoked in response to some arbitrary condition occurring. They correspond only loosely to schemata. Schemata are larger-grained than KS's, and capable of being interrupted. Schemata also serve to cluster actions to be taken to achieve a goal; many KS's, on the other hand, may be needed to achieve a single goal. The use of schemata should allow the reasoner to behave in a manner that a reasoner can understand: e.g., question-asking should be more focused. Schemata should also facilitate explanation, since the actions taken to achieve a goal, though possibly temporally disjoint, can still be explained in relation to one another.

The VISIONS Schema System (Weymouth, 1986) uses an approach similar to ours for interpreting visual scenes. Their schemata are specialists in particular vision tasks and can work in parallel to interpret a scene. Unlike our schemata, theirs are largely represented using procedures written in a programming language; it is therefore not possible for their program to reason about or modify their schemata, as is potentially possible using our representation of schemata. In addition, parallel execution of schemata is not feasible in our domain, given the goals of focused question-asking and modeling a human diagnostician's behavior.

The NASL (McDermott, 1978) program concentrated on the interaction of planning and execution. In many respects, our schemata are similar to NASL's tasks: both are hierarchical, bottoming out at the primitive action or primitive task level. However, our schemata are somewhat more flexible than NASL's tasks, and we make explicit use of goals; the latter allows the potential of specifying the goal of a task without necessarily specifying the steps to achieve it, thus allowing the reasoner to make such decisions at run-time.

We face some of the same problems as did NASL, too, in the chore of selecting which schemata to pursue at each point in problem solving. NASL made use of choice rules, which contained the strategic knowledge of that system. Our strategic schemata can be viewed in the current implementation as packages of such choice rules. However, the ultimate goal is to make them less rule-like and more schema-like, specifying steps for the reasoner to perform in order to select schemata to apply.

Firby (1987) is also concerned with interleaving planning and execution in environments that change during planning. The behavior of his RAP planner is quite similar to that of our reasoner. However, RAPs would seem to be somewhat more simple than our schemata, and oriented towards real-time control rather than diagnosis. In addition, we extend the idea of using packets of control knowledge to the meta-level by using strategic schemata to direct the reasoner's attention.

CONCLUSION

Medical diagnosis can be fruitfully viewed as a planning task in which planning is interleaved with diagnosis. New information may be discovered during diagnosis which should impact the future problem-solving behavior of the diagnostician. The diagnostician must be *opportunistic* in order to take notice of and respond to this new information as it becomes available.

Schema-based reasoning provides one approach to opportunistic reasoning. By representing problem-solving knowledge as packets of procedural information designed to achieve a goal, the reasoner can activate schemata as goals arise due to changes in the environment. When several schemata are active, the reasoner selects the one to apply based on the reasoner's current focus of attention—i.e., the goal the reasoner is trying to achieve. Goals are selected by the reasoner based on information from two sources: general goal-ordering information, stored in strategic schemata; and specific goal-ordering informa-

[8] Again, using a method similar to that of INTERNIST-1.

tion, stored in the dxMOP representing consultations similar to the current one.

Schemata are flexible, and enhance the reasoner's ability to respond to changes in the environment in two ways: (1) the order of their steps need not be completely determined—this allows the reasoner to select the next step of a schema based on the state of the world resulting from the application of the previous step; and (2) steps may specify goals, which the reasoner can attempt to satisfy at run-time by retrieving schema specific to the current situation. This flexibility, plus its opportunistic character, allows schema-based reasoning to be viewed as a type of reactive planning (e.g., Firby, 1987).

Though this research addresses medical diagnosis, we believe that schema-based reasoning can be usefully applied to other tasks. Our approach should be useful for any task in which planning and execution must be interleaved, or in which all features of the problem cannot be known at the start of the problem.

ACKNOWLEDGMENTS

Many thanks to Janet Kolodner, Hong Shinn, and Elise Turner for their comments on earlier drafts of this paper, and to our domain expert, Eric Honig, of Emory University and Grady Memorial Hospital.

REFERENCES

Agre, P.E., and Chapman, D. (1987). Pengi: An implementation of a theory of activity, in *Proceedings of the Sixth National Conference on Artificial Intelligence*, pp. 268–272.

Cullingford, R.E., and Kolodner, J.L. (1986). Interactive advice giving. In *Proceedings of the 1986 IEEE International Conference on Systems, Man, and Cybernetics.*

Davis, R., and Buchanan, B.G. (1984). Metaleval knowledge. In B.G. Buchanan and E.H. Shortliffe (eds.), *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project.* Addison-Wesley Publishing Company, Reading, Massachusetts, pp. 507–530.

Erman, L.D., Hayes-Roth, F., Lesser, V.R., & Reddy, D.R. (1980). Then HEARSAY-II speech-understanding system: Integrating knowledge to resolve uncertainty, *Computing Surveys*, Vol. 12, No. 2.

Firby, R.J. (1987). An investigation into reactive planning in complex domains, in *Proceedings of the Sixth National Conference on Artificial Intelligence*, pp. 202–206.

Gomez, F., and Chandrasekaran, B. (1982). Knowledge organization and distribution for medical diagnosis. In W.J. Clancey and E.H. Shortliffe (Eds.), *Readings in Medical Artificial Intelligence*, pp. 320–338. Reading, Massachusetts: Addison-Wesley Publishing Company, 1984. (Originally published in *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-11, No. 1, pp. 34–42 (1981).)

Hayes-Roth, B. (1985). A blackboard architecture for control, *Artificial Intelligence*, Vol. 26, No. 3, pp. 251–321.

Kolodner, J.L. (1985). Experiential processes in natural problem solving. Technical Report #GIT-ICS-85/123, School of Information and Computer Science, Georgia Institute of Technology, Atlanta, Georgia.

Kolodner, J.L., and Kolodner, R.M. (1987). Using experience in clinical problem solving: Introduction and framework. In *Proceedings of the 1987 IEEE International Conference on Systems, Man, and Cybernetics.*

McDermott, D. (1978). Planning and acting, *Cognitive Science*, vol. 2, pp. 71–109.

Miller, R.A., Pople, H.E., Jr., and Myers, J.D. (1982). INTERNIST-1, an experimental computer-based diagnostic consultant for general internal medicine, *New England Journal of Medicine*, vol. 307, pp. 468–476.

Schank, R.C. (1982). *Dynamic Memory*, Cambridge University Press, New York.

Schank, R.C., and Abelson, R. (1977). *Scripts, Plans, Goals and Understanding*, Lawrence Erlbaum Associates, Hillsdale, NJ.

Turner, R.M. (in press). Organizing and using schematic knowledge for medical diagnosis. Submitted to the Seventh National Conference on Artificial Intelligence (and also appearing in the 1988 DARPA Workshop on Case-based Reasoning).

Weymouth, T.E. (1986). *Using Object Descriptions in a Schema Network for Machine Vision*, Technical Report 86–24 (Ph.D. thesis) Dept. of Computer and Information Science, Univ. of Massachusetts.

Integrating Case-Based and Causal Reasoning

Phyllis Koton[1]
Laboratory for Computer Science
Massachusetts Institute of Technology

Introduction

Much research in artificial intelligence has been directed toward the development of high-performance domain-specific problem solving systems, called *expert systems* or *knowledge-based systems*. Many current expert systems rely on *associational* knowledge (heuristics, empirical associations, "rules of thumb") for their reasoning power. An alternative approach, *causal reasoning*, uses a model of the objects in the domain and the operations that can be performed on those objects. Causal models can provide richly-detailed knowledge bases for reasoning in many domains, but models are inefficient compared to the associational knowledge typically used in expert systems.

Human problem solvers are able to use both associational and causal reasoning. We recognize and quickly solve common problems, but can use more detailed causal knowledge when faced with novel or difficult problems. An artificial reasoning system that combined both types of knowledge, using associational knowledge for speed, and reserving the ability to reason from a model when necessary, similarly, would be highly desirable. Case-based reasoning techniques [Kolodner, 1985] can be used to improve the performance of causal model-based systems, because the act of retrieving a similar case and using its solution is clearly associational: features of a problem are associated with a solution to that problem. However, case-based reasoning techniques have not been widely used to enhance the performance of expert systems. Also, case-based reasoning systems suffer from lack of a model. There is no justification (other than coincidence) for believing that transferring a solution from a previous case to a new case will produce a valid solution. The combination of case-based reasoning techniques with causal reasoning could result in substantially improved expert systems.

Overview of CASEY

I have developed a program, CASEY, which integrates case-based and causal reasoning. The causal reasoning component employs a model of the cardiovascular system developed for the Heart Failure program [Long, et al, 1986], an expert system for managing patients with heart disease. The case-based reasoning component uses a self-organizing memory system [Kolodner, 1983] to store descriptions of all patients the program has seen, and generalizations derived from similarities between the patients. The patient description is comprised of *features*. These include both input data, such as signs and symptoms, test results, history and current therapy information, and solution data, such as the causal explanation for the patient, the diagnosis, therapy recommendation and outcome information.

CASEY's output is a causal explanation that describes a relationship between physiological states in the model and observable features of the patient. This is produced using a five-step

[1]The work reported here has been supported (in part) by National Institutes of Health grants R01 LM 04493 from the National Library of Medicine and R01 HL 33041 from the National Heart, Lung, and Blood Institute. Robert Jayes, MD kindly provided the example cases. William Long's Heart Failure program provided an excellent resource for this work. Peter Szolovits, Ramesh Patil, and William Long gave much helpful advice in supervising this research.

process. First, CASEY finds a case similar to the new patient in its case memory. Next, it evaluates the significance of any differences between the new case and the retrieved case. During this phase the match can be invalidated if there are significant differences. If none of the differences invalidate the match, CASEY adapts the solution from the retrieved case to fit the new case. If a match is ruled out, or if no similar previous case is found, CASEY uses the Heart Failure program to produce a solution for the case *de novo*. The new case and its solution are stored in CASEY's memory for use in future problem solving.[2] Finally, the features which were causally important in the solution of this problem are noted in the memory.

Retrieving, adapting, and storing cases are standard procedures of a case-based reasoner. Because the match between a new problem and a previously solved problem usually is only partial, there may be differences between the two cases that preclude using even a modified version of a retrieved solution for a new problem. The justification step ensures that a retrieved solution can be supported by the data in the new problem. Evaluating the features of the new problem to determine which were important to the solution helps the program make better matches in the future, because it allows the program to distinguish between random features and important ones.

CASEY differs from previous case-based reasoning systems because it uses information from its causal model to effect these steps. During retrieval, causal knowledge is used to select important features of the new case for matching. During justification, causal reasoning is used to judge the significance of differences between the new and previous cases. In repairing a retrieved solution, the causal model determines what changes should be made in the retrieved solution so that it fits the present case. Feature evaluation uses the causal explanation of the new case to determine its important features.

Matching and Retrieval

Most case-based reasoners use a similarity metric to gauge the similarity of two problems, and to choose the best match for the current problem. Similarity metrics use a combination of the number of features in common and the relative importance of those features. In many case-based reasoning systems, the relative importance of features for matching is predetermined by the system designer (for example, [Simpson, 1985], [Hammond, 1986], [Bain, 1986]). Another approach is to determine the important features based on a program's experience of what was important in solving similar problems.

CASEY matches a new case against cases in its memory using *every* feature in the patient description. However, not all the features are equally important in matching a case to a previous case. Furthermore, the important features for matching may vary from case to case. Therefore, CASEY determines the important features for matching dynamically for each new case presented to the system, and gives these features greater weight for matching. Important features are defined as those that played a role in the causal explanation of previous similar cases.

Figure 1 shows a sample patient presented to CASEY, Oprah. The retrieved case for Oprah, a patient named Mary, is shown in Figure 2.[3] The features marked with an asterisk in Mary's description are those that were used in the solution to her case. Oprah has some, but not all, of the features that were important in the case of Mary. The two cases also share some features that were

[2]The user has the option of rejecting CASEY's solution, in which case Heart Failure program is used to produce a causal explanation, which will be stored in memory.

[3]The patient descriptions in these illustrations have been simplified to conserve space by excluding features with normal values. A patient description typically consists of about 40 features.

```
(DEFPATIENT "Oprah"
HISTORY                        PHYSICAL-EXAM
(AGE . 77)                     (APPEARANCE NO-APPARENT-DISTRESS)
(SEX FEMALE)                   (PULSE SLOW-RISE)
(DYSPNEA ON-EXERTION)          (AUSCULTATION MURMUR S2)
(CHEST-PAIN ANGINAL)           (MURMUR SYSTOLIC-EJECTION-MURMUR)
(ANGINAL UNSTABLE)             (S2 SOFT-A2)
VITAL-SIGNS                    (APEX-IMPULSE SUSTAINED)
(BLOOD-PRESSURE 147 89)        LABORATORY-FINDINGS
(HEART-RATE . 84)              (EKG LVH NORMAL-SINUS)
(RESP . 14)                    (CXR CARDIOMEGALY)
(TEMP . 98.6)                  (CARDIOMEGALY LV))
```

Figure 1: Description of patient Oprah.

```
(DEFPATIENT "Mary''
HISTORY                                PHYSICAL-EXAM
(AGE . 67)                             (APPEARANCE DIAPHORETIC* ANXIOUS*)
(SEX FEMALE)                           (PULSE NORMAL)
(DYSPNEA ON-EXERTION*)                 (AUSCULTATION MURMUR S2)
(CHEST-PAIN ANGINAL)                   (MURMUR SYSTOLIC-EJECTION-MURMUR*)
(ANGINAL UNSTABLE* EXPERIENCING*)      (S2 SINGLE*)
VITAL-SIGNS                            (APEX-IMPULSE SUSTAINED*)
(BLOOD-PRESSURE 148 90)                LABORATORY-FINDINGS
(HEART-RATE . 99)                      (EKG NORMAL-SINUS LV-STRAIN*)
(RESP . 14)                            (CXR CARDIOMEGALY)
(TEMP . 98.7)                          (CARDIOMEGALY GENERALIZED*))
```

Figure 2: Description of patient Mary.

not used in the solution of Mary's case. CASEY matched Oprah to Mary using the *similarities* between the two cases. It now calculates the *differences* between the two cases (shown in Figure 3) and passes this information to the justifier, along with the solution retrieved from Mary's case.

Justification and Adaptation

Two cases might have many similar features yet have one critical difference that invalidates the match. The critical question is whether different values of features in the problem description still support the same solution. In CASEY's domain, this manifests itself as whether different patient symptoms still support the same causal explanation. CASEY therefore uses a set of *evidence principles* to evaluate differences between the new case and a retrieved case by examining the relationships between evidence and physiological states in the Heart Failure model. These principles rely on such concepts as alternate lines of evidence for states, additional supporting evidence for states, and inconsistent evidence. The module in CASEY that performs this evaluation is called the *justifier*.

A difference is *insignificant* if it does not affect the retrieved causal explanation. For example, the difference between Mary's and Oprah's temperature is insignificant because both temperatures are normal. A difference is said to be *repairable* if the features of the new case can be fit to the retrieved causal explanation. Consider, for example, the fragment of Mary's causal explanation shown in

Feature name	Value for **Mary**	Value for Oprah
age	67	77
anginal	unstable* experiencing*	unstable
blood-pressure	148/90	147/89
heart-rate	99	84
temp 98.7	98.6	
appearance	diaphoretic* anxious*	normal
pulse	normal	slow-rise
s2	single*	soft-a2
ekg	lv-strain*	lvh
cardiomegaly	generalized*	lv

Figure 3: Differences between patients Mary and Oprah.

Figure 4. The feature EKG: LV STRAIN is evidence supporting the state LV HYPERTROPHY. Oprah does not have lv strain on her EKG. CASEY looks in the causal model for other features that are evidence of LV HYPERTROPHY, and determines whether Oprah has any of those features. In fact, Oprah has two such findings: EKG: LVH and LV CARDIOMEGALY. Therefore, CASEY can justify keeping this state in Oprah's causal explanation. If all differences between the new case and the retrieved case are insignificant or repairable, then the transfer of solutions from the precedent to the current case proceeds.

In case-based reasoning systems without causal models, the problem solver finds the best match, transfers its solution to the new case, and hopes for the best. Sometimes a retrieved case leads the problem solver down the wrong path. Evaluating differences by use of a causal model improves the likelihood that the retrieved solution applies to the new case. When CASEY justifies the match between the old case and the new case, it demonstrates that although there are differences between the cases, the causal model still supports the retrieved solution.

Modifications to the solution are necessary for partial matches between cases. *Repair strategies* are invoked by the justifier when it discovers a repairable difference between the new case and the retrieved case. Repair strategies adapt a previous solution to a new case by adding or removing nodes and links on a copy of the retrieved causal explanation. In the example of Mary and Oprah, the justifier would invoke causal repair strategies to remove the evidence EKG: LV STRAIN from the state LV HYPERTROPHY, and add the evidence EKG: LVH and LV CARDIOMEGALY to the list of evidence supporting that state. Repaired solutions do not have to be tested (as is required in, e.g., [Hammond, 1986], [Simmons & Davis, 1987]) because the validation of the solution by the causal model has already taken place in the justification phase.

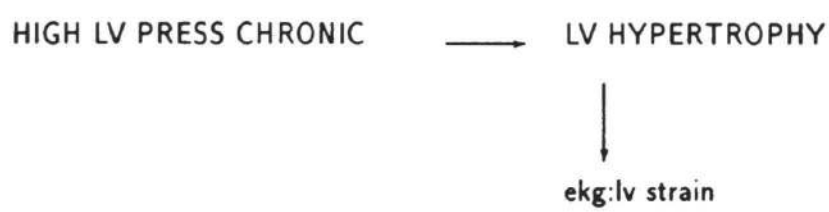


Figure 4: A fragment of Mary's causal explanation.

Storage

The use of causal reasoning influences the way a new case is stored in the case memory. A cases is indexed both by the input features that describe the case and the causal explanation that was derived for the case. CASEY also makes generalizations about the cases it has solved by finding similarities between the new case and cases already in its memory.

Generalizing the patient descriptions allows CASEY to make predictions about patients who share features [Kolodner, 1985] by recognizing co-occurrences. CASEY generalizes *all* the features in the patient description, not just the causally-related features. The Heart Failure model is incomplete, so it is possible that certain non-causal features are related to (and therefore can predict) some cause not represented in the model. The sex of the patient is an example: no state in the causal model uses the sex of the patient as evidence, yet there are causal relationships between gender and heart disease. The age of the patient is another example. The use of case-based reasoning therefore allows CASEY to *improve* on the performance of the Heart Failure system by learning new associations between features and solutions. At the same time, making generalizations about groups of similar patients reduces the effect of noise (random, unimportant features in the patient description) on the performance of the program. This is because spurious features are likely to occur randomly, whereas important features will tend to recur with some regularity in cases presented to the program.

Generalizing the explanations produces partial explanations that explain the features that the new case and retrieved case have in common. This allows CASEY to produce a partial solution for a similar problem in the future even in the absence of enough information to explain the whole problem.

Feature Evaluation

The use of a causal model is essential to CASEY's feature evaluation step. After producing the causal explanation for a new case, CASEY determines the important features of the *new* case (those features which were used as evidence in the causal explanation) and increments the weights of these features in the case memory. For example, after the case of Oprah is solved, the features EKG: LVH and LV CARDIOMEGALY are given extra weight in the memory because the causal model says that these features will be useful in identifying future patients with LV HYPERTROPHY, i.e. these features predicted a part of the solution. Determining the importance of features by experience is reasonable because the usefulness of a feature cannot always be determined in advance. This also allows the problem solver to adapt to changes over time in the types of problems is is presented. Giving extra weight to causally-related features is reasonable because because causality often indicates which features are important in the case for matching [Winston, 1981], [Schank, 1986].

CASEY also identifies the states in the causal explanation of the new case that are directly linked to findings, and stores the new case in memory using these states as indices. Future cases that contain evidence to support these states will retrieve Oprah's case as a match.

Related Work

CHEF [Hammond, 1986] combines case-based reasoning with a simple causal model. Therefore its causal reasoning can consist solely of chaining rules backward from an observed failure to a

cause. This would not scale up to a reasonably sized domain. Recent work by Resnick and Davis [Resnick & Davis, 1988] combines a memory of past cases with explanation-based generalization of a causal model to produce a generalized description of a hardware fault, but their technique requires an exact match between the new problem and the old problem. IVY [Hunter, 1987], like CASEY, uses information of what was important to its reasoning task to select important features of the problem for storage. IVY, however, simplifies the selection problem by using heuristics to dispense with most of the features presented to the program.

Other systems have combined reasoning from a causal model and associational reasoning. ABEL [Patil, 1981] maintained a description of the patient at five levels of detail. It did not have a learning component. The Generate, Test and Debug method [Simmons & Davis, 1987] always uses its causal model to test proposed hypotheses, and always generates them using associational rules. CASEY *decides* when to use associational or causal knowledge. It determines whether the new case is sufficiently similar to ones it has already solved to use the associationally-derived solution. If not, it resorts to causal reasoning.

Discussion and Conclusions

CASEY *integrates* causal and case-based reasoning techniques in a program which is efficient, can learn from its experiences, and solves commonly-seen problems quickly, while maintaining the ability to reason using a detailed knowledge of the domain when necessary. The causal component is enhanced by the ability of the case-based component to learn new associations and compile detailed reasoning structures into simple associations between features and solutions. The case-based component is improved by the use of a causal model because the model can demonstrate that a retrieved solution will be helpful for a new case, and the model can be used to identify important features for matching.

Since determination of important features is based on information in the causal model, it is reasonable to ask why the Heart Failure model is not simply "compiled" to produce all this information in the form of associational rules relating important symptoms and physiological states. In fact, that is exactly what CASEY is doing, but it is compiling the knowledge incrementally, associating features of problems with solutions for the cases it has seen. Also, the Heart Failure program can generate solutions involving multiple diagnoses, its model provides the relative importance of features only for single diagnoses. To compile all of the Heart Failure program's knowledge taking into account multiple diagnoses would be computationally intractable. Because CASEY also makes generalizations about patients who have multiple diagnoses, it can create associational knowledge relating features to solutions involving multiple diseases.

CASEY's most serious limitations is that it assumes that there is very limited interaction between entities in the underlying causal model. For example, it has no concept of two states jointly causing a third. This weakness is due to the fact that no such interactions are represented in the Heart Failure model. This limits the generality of CASEY's reasoning techniques to other systems which also make this assumption. CASEY's evidence principles currently are being extended to handle more complex interactions.

Even if CASEY cannot solve a new problem completely because it lacks sufficient input information, it can often give a partial solution. Similarly, when CASEY is given a problem at the boundary of its expertise, it can produce a partial solution for features that are handled by its causal model, and leave the remaining features unexplained. Therefore, techniques such as are used in CASEY might be useful in integrating several causal models for different domains. Each

model might use those features which it could explain, leaving the other features for other models. This is a topic for future work.

References

[Bain, 1986] William M. Bain. A case-based reasoning system for subjective assessment. In *Proceedings of the National Conference on Artificial Intelligence*, pages 523–527, American Association for Artificial Intelligence, 1986.

[Hammond, 1986] Kristian Hammond. *Case-based Planning: An Integrated Theory of Planning, Learning and Memory*. PhD thesis, Yale University, 1986.

[Hunter, 1987] Larry Hunter. Indexing with knowledge goals: how to put knowing what you know to use. In *Proceedings of the Forth International Workshop in Machine Learning*, 1987.

[Kolodner, 1985] Janet L. Kolodner. *Experiential Processes in Natural Problem Solving*. Technical Report GIT-ICS-85/22, School of Information and Computer Science, Georgia Institute of Technology, 1985.

[Kolodner, 1983] Janet L. Kolodner. Maintaining organization in a dymanic long-term memory. *Cognitive Science*, 7:243–280, 1983.

[Kolodner, 1988] Janet L. Kolodner. Retrieving events from a case memory: a parallel implementation. 1988. Submitted to AAAI88.

[Long, et al, 1986] W. J. Long, S. Naimi, M. G. Criscitiello, and R. Jayes. Using a physiological model for prediction of therapy effects in heart disease. In *Proceedings of the Computers in Cardiology Conference*, IEEE, October 1986.

[Patil, 1981] Ramesh S. Patil. *Causal representation of patient illness for electrolyte and acid-base diagnosis*. TR 267, Massachussetts Institute of Technology, Laboratory for Computer Science, 545 Technology Square, Cambridge, MA, 02139, October 1981.

[Resnick & Davis, 1988] Paul Resnick and Randall Davis. Improving performance of model-based reasoning by learning from experience. March 1988. Submitted to AAAI88.

[Schank, 1986] Roger C. Schank, G. C. Collins, and Larry Hunter. Transcending inductive category formation in learning. *Behavioral and Brain Sciences*, 9:639–686, 1986.

[Simmons & Davis, 1987] Reid Simmons and Randall Davis. Generate, test, and debug: Combining associational rules and causal models. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, pages 1071–1078, 1987.

[Simpson, 1985] Robert L. Simpson. *A Computer Model of Case-Based Reasoning in Problem Solving: An Investigation in the Domain of Dispute Mediation*. Technical Report GIT-ICS-85/18, Georgia Institute of Technology, 1985.

[Winston, 1981] Patrick H. Winston. *Learning New Principles from Precedents and Exercises: The Details*. AIM 632, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, 545 Technology Square, Cambridge, MA, 02139, 1981.

Modeling Software Design Within a Problem-Space Architecture

Beth Adelson
Tufts University

Introduction

In this paper[1] we describe research on modeling software design skills within the Soar problem-solving architecture (Laird, Newell & Rosenbloom, 1987). We focus on an analysis of software designers designing an electronic mail system. The research addresses the issues of: 1. Guiding mental simulations of a design-in-progress using learned schemas. 2. The interaction of general knowledge of design with domain knowledge about the system being designed. 3. Progressive deepening of problem representations during problem-solving (Newell & Simon, 1972; deGroot, 1965).

Below we describe Soar, the theory which underlies the theoretical perspective of the research described here. We then discuss our protocol data on software design. This is followed by a description of the Soar-based system suggested by the data.

Soar: A General Architecture for Cognition

Ultimately, the Soar architecture is intended to embody a *unified theory of cognition*; capable of accounting for a range of cognitive problems or "tasks". Additionally, it is expected to be able to do so by relying on the mechanisms of recursive sub-goaling and chunking.

Currently, Soar can solve a wide range of standard AI problems. It can solve most of the "toy" problems such as eight puzzle and tower of Hanoi (Laird & Newell, 1983) which require goal-oriented action, without requiring knowledge about the problem domain. It can also solve knowledge-intensive, expert system tasks, such as those solved by R1 and Neomycin. A variety of search strategies have been implemented as well as weak method problem-solving strategies such as generate and test and means-ends analysis. Soar also exhibits learning learning with practice, transfer across tasks and generalization.

In terms of accounting for tasks that are central for a theory of cognition, research is now being conducted to look at learning by analogy and reasoning with mental models (Golding,1988; Polk, 1988; Steier & Kant, 1985; Steier & Newell, 1988). The research presented here is intended to be part of this effort to extend the range of Soar's performance to complex cognitive tasks.

The Nature of Problem-Solving in Soar

In Soar problem-solving is characterized as movement through successive *states of knowledge* in a *problem space* in order to achieve a *goal* (Newell, Shaw, & Simon, 1960; Newell & Simon, 1972; Card, Moran & Newell, 1980; Newell, 1980). The problem-solver starts out in an initial state which contains an incomplete representation of the problem solution and a *description* of what would constitute a sufficient solution. The description of the solution could be, for example, the desired *behavior* for the electronic mail system designed here, whereas the solution *itself* would be a *pseudocode* specification of the *mechanism* producing the behavior.

The problem solver's relevant knowledge is then brought to bear and the initial representation of the problem is transformed in a way that brings it closer to the goal state representation; the problem solution. Relevant knowledge may consist of specific information about the problem domain as well as general problem-solving strategies.

[1] We are grateful to David Steier for his continuing generous help. This work was supported by grants from the Design, Manufacturing and Engineering Program and the Knowledge and Data Base Systems Program at NSF.

2	8	3
1	6	4
7	b	5

2	8	3
1	b	4
7	6	5

1	2	3
8	b	4
7	6	5

Figure 1: Starting State (left), Second State (center) and Goal State (right) for the Eight Puzzle. (The "b" represents the blank cell.)

Elements of the Architecture

To model problem-solving as it is framed above we need to be able to provide accounts of: the representation of the current problem solution at varying stages of completion; the representation of whatever is known about the desired problem solution; and knowledge about how to assess and transform the partial solution with regard to the desired solution.

The above are realized using the following elements of the Soar architecture:

- Production Memory (PM). This encodes the long-term knowledge that is needed during problem-solving. This can include factual knowledge about the problem being solved, strategic knowledge about how to proceed in problems like the current one and operational knowledge about specific problem-solving moves to be made in a given situation. It is the use of this operational knowledge that transforms the problem solution from an inital version into a goal state version. The productions that contain this type of operational knowledge place operators into working memory. When these operators are applied they transform the solution-in-progress (Section).
- Working Memory (WM). This holds the representations of the current and desired problem solution. WM also holds long term knowledge that has been identified as relevant.
- The Decision Cycle. This brings the appropriate knowledge in production memory to bear given the state of things in working memory. The difference between the starting and goal states is reduced through the decision cycle. The decision cycle is made up of two phases:
  1. An *elaboration* phase that causes already known information in production memory to be added to working memory. Information in production memory is added to working memory if it is relevant to what presently is in working memory. Elaboration is acheived by a matching process. The antecedents of all productions in PM are matched against the contents of WM; all productions that do match "fire" causing the objects described in the productions' consequents to be placed in WM.
  2. A *decision* phase that makes problem-solving decisions based on the information in working memory. The decision process begins once the elaboration process has added all that it currently can to working memory.

Using the Architecture: An Example of Problem-Solving in Soar

Below we present a description of Soar solving the eight puzzle (Figure 1). The example illustrates how the elements of the architecture function in order to move the problem-solver through the problem space towards the goal state. The eight puzzle is chosen here *not* as a representative cognitive task, but because its simplicity allows us to focus on the Soar architecture.

Soar begins by moving the 6 down into its desired spot. This occurs because of the elaboration and decision processes acting in turn. First, during elaboration, productions fire and **down**, **right** and **left** are placed in WM and marked as acceptable candidate operators. Additionally, a means-ends analysis production fires

marking **down** as best, because it moves the 6 into its desired place. Next, during the decision cycle 6 is chosen as the best move. This allows the move to actually be made during the following cycles.

Next, while trying to make the second move, **down**, **right** and **left** are all placed in WM and all marked as acceptable candidate operators. This causes the decision process to reach a "tie impasse". The information in production memory about this problem space is incomplete; it cannot resolve the tie. However, here is where the notion of sub-goals and problem spaces comes into play.

Sub-goaling to resolve impasses

When an impasse is reached the Soar architecture sets up a sub-goal to resolve the impasse. Here we see a tie impasse, however, no-change, conflict and rejection impasses are also possible.

For a tie impasse in the eight puzzle the sub-goal is to "Select" an operator from the set of possible ones. This is achieved by moving into the *selection problem space*. A further sub-goal results in which the candidate operators are actually tried out and the state that will result from each one is evaluated. For this example, **down** is found to be best because it moves the 6 into place whereas **left** and **right** move the 5 and 7 out of their desired spots.

Three points are important here: 1. The detection of the impasse and the setting up of the appropriate type of sub-goal is *not* done by the task specific eight puzzle productions; it is done by the architecture. 2. Once a sub-goal is established it is pursued and resolved in the *same* way as a higher level goal. A problem space is selected; a current and goal state are defined; and operators are then applied to the current state in order to transform it into the goal state. 3. This sub-goaling can occur to an arbitrary depth. These three points lead to some of the appeal that Soar has as a theory. By being able to detect impasses and set up appropriate sub-goals, the architecture, the part of Soar which is specified in advance and remains constant across tasks, does a good deal of the problem-solving. Additionally, the ability to solve problems in this uniform way (by recursive sub-goaling) allows Soar to provide a parsimonious account of complex problem solving.

Turning to software design we will see that organizing the problem-solving into problem spaces continues to be useful. We will also look at the way in which the problem spaces are related and how information from one problem space can further problem-solving in another.

Modeling Software Design within Soar

Method

Below we present our data on software design.
Subjects. Three expert software designers served as subjects.
Procedure. We presented each of the designers with the following design task to work on.

> Design an electronic mail system around the following commands:
> **READ, REPLY, SEND, DELETE, SAVE, EDIT,** and **LIST-HEADERS.**
> The goal is to get to the level of *pseudocode* that could be used by professional programmers to produce a running program. The mail system will run on a very large, fast machine so hardware considerations are not an issue.

Analysis of the Protocol Data

Generally protocol data can be seen as a series of episodes, with each episode reflecting the single, current focus of the subject's attention.

Pairs of episodes
One striking aspect of the protocol discussed here is that the episodes formed related pairs. The first episode

Episode 1. View the system as a set of functions.
Episode 2. Simulate the behavior of the system's functions. The commands prepare, send receive and store must be included in specifying the mailer's functionality. Discover that error recovery must be handled gracefully throughout the system.

Episode 3. View the system as a set of data objects.
Episode 4. Elaborate the features of the data objects . In the mailer messages are the data objects. Messages have destinations of senders and receivers. Additionally, messages are grouped together in stores. The stores can have various functions. For example the mail system needs a store for messages that the user has received but not yet read, as well as a store for messages that have been read but not yet saved or deleted.

Episode 5. View the system as a set of concurrent functions.
Episode 6. Simulate the behavior of a system in which there are concurrent senders and receivers. Discover that the design needs to specify when users should be notified that new mail has arrived (as it arrives, only at log on, etc). Also discover that, since mail is both being sent and received, more than one type of processing must be handled and therefore a "dispatch demon" is needed to handle the flow of messages.

Episode 7. View the system as a state machine for the states of a user.
Episode 8. Simulate the behavior of the system as a scenario in which the user logs on and issues a sequence of mail commands. He is notified that he has mail, he lists the headers, he reads a message, he makes some disposition of the message and then is able to begin again (listing headers, etc.). The designer discovers that the post-conditions of the commands need to be enumerated both to refine the command definitions and to understand the potential interactions between commands. For example, if READ includes an implicit and immediate DELETE it will prevent the user from being able to save or forward the message.

Episode 9. View the system as a state machine for the states of messages.
Episode 10. Simulate the behavior of the system in terms of the states of the messages. A message is created, sent, received, read and disposed of. These actions described at the level of files and locations within files are sufficient to generate pseudocode.

Table 1: Description of Episodes 1 through 10.

in a pair appears to take place in a general design space and the second episode appears to take place in a space containing knowledge about mail systems. The first two episodes from S1 illustrate this phenomenon.

> Episode 1: S: "....I'm going to start working here, functions of an electronic mail."
> {writes 'Functions' and 'Data' in two separate columns}
> Episode 2: "We must be able to: Prepare, Send, Receive...
> {writes prepare, send, receive under the heading 'Functions'}
> ...the system must be able to store them,
> the system must be able to handle abnormalities throughout it."

In episode 1 S1 decides to view the system as a set of functions. In episode 2 he goes on to enumerate what those functions would be.

In Table 1 we present a summary of episode pairs for the first 10 episodes of S1's protocol. The first episode in each pair establishes the goal of viewing the sytem from a particular *perspective*. The second episode instantiates the view as a simulation.

Schemas

The design process seems driven by an experience-based schema for two reasons: First, successive episodes do not appear to arise from the context that immediately precedes them; in episodes 3 and 4 the system is viewed as a set of data objects and then in episodes 5 and 6 as a set of concurrent processes. Second, the particular views chosen, such as dealing with concurrency issues, would be ones to develop given these designers' experience with communications systems.

The structure of the schema is also interesting; taken in order the five views comprise a set that would be effective in uncovering most of the aspects of the system that need refinement. The first view looks at the commands of the mailer, the second view at the messages themselves. Once the commands have been specified it becomes possible to look at the interactions produced when they are used in sequence. This is

uncovered by the fourth view which looks at the system as a state machine from the user's perspective. It also becomes possible to look at their concurrent functioning (view 3). The fifth view looks at the interaction of commands and messages; it therefore, is dependent upon having specified the commands and messages in the first and second views. Both the pairing and the ordering of the episodes is explained by the use of an underlying schema.

Interaction of Domain and General Knowledge

In designing a large software system the designers employ three types of knowledge; general knowledge of *design*, knowledge for representing *systems as pieces of pseudocode* and domain specific knowledge of how a mail system *behaves*. From our perspective, these bodies of knowledge can be seen as three *problem spaces*; a *design* space, a *pseudocode* space and a *mail* space. In simulating views of the system's behavior there has to be a mapping between the designer's high-level problem space for design, in which the schema resides and the *domain* space where knowledge about the *behavior* of mailers resides; it is in this domain space that the view is instantiated, run, and evaluated. For example, in episode 7 the designer chooses to view the system as a state machine in which the user goes through a sequence of state transitions. This gives rise to episode 8, in which the designer instantiates this state machine by constructing a simulation in which the user logs on, is notified that he has mail, lists the headers, reads and saves a message and then begins again. This means that the user's state transitions need to be put into correspondence with the issuing of commands such as **READ** or **SAVE**. Additonally, data objects must be understood to correspond with messages.

In the domain space there needs to be enough knowledge about the behavior of the mail commands to propose and simulate candidate versions of them. The candidate versions then have to be evaluated by comparing their behavior to some representation of the ideal behavior. The candidates can then be modified in accord with the results of the evaluation. The ability to modify a representation of a command in pseudocode space based on the results of a simulation in mail space implies a mapping between mail space and pseudocode space.

Progressive deepening

Progressive deepening is the retracing of steps along a previously taken problem-solving path (Newell & Simon, 1972; deGroot, 1965). The retracing is done because the first trip down the path was not sufficient; and new, relevant information has been acquired.

In episodes 2 through 10 (Table 1) we see the progressive deepening of the mailer's commands. In episode 2 the commands are just listed as a set of functions to be specified. In episode 8 the designer simulates these same commands using each one's output as input to the next and discovering that the side-effects of each have to be elaborated. For example, he decides that a user should be able to list all the message headers without being committed to then reading them. In episode 10 the designer finally simulates the commands at a level sufficient to generate pseudocode. Here he uses language that refers to reading from and writing to files.

Simulation *and* progressive deepening arise naturally within a Soar architecture. There are two reasons why simulation occurs within Soar. The first reason has to do with seeing complex problem solving as occuring in a *set* of related problem spaces. Problem-solvers have different *types* of representations of the problem solution in different problem spaces. Additionally, the information contained in one type of representation may contribute to the development of another. In design, simulations arise because detailed information about the behavior of a mail command, obtained from a simulation in *mail* space, can help in developing the representation of the command in *pseudocode* space.

The second reason for simulation has to do with comparing current and goal states. In designing the mail system, the goal state is described in terms of the desired *behavior* of the mail system. However, the problem solution is a pseudocode description of the mailer. In order to compare this pseudocode representation to the goal the pseudocode must be simulated.

The simulations done by the designers need to go through progressive deepening; at the beginning of the design session the designers' representation of the mailer is in terms of the high level *behavior* of the system.

Design Space Operators:
The operators causing differing views of the system to be taken.

1. Design the system as a set of functions.
2. Design the system as a set of data objects.
3. Design the system as sets of concurrent functions.
4. Design a state machine of the states a user goes through.
   (Resulting in a focus of attention on interactions between functions.)
5. Design a state machine of the states a message goes through.
   (Resulting in a focus of attention on interactions between function and data.)

Pseudocode & Mail Space Operators:
The operators to generate, run and evaluate candidate versions of the functions' mechanisms in pseudocode space and the functions' behavior in mail space.

1. Prepare.
2. Send.
3. Receive.
4. List-Headers.
5. Read.
6. Store.
7. Delete.

Table 2: Operators in the Design and Mail Spaces

This is the representation that would be likely for a person who had used, but not actually designed such a system. The representation of the behavior needs to be refined to a degree that allows that behavior to be expressed as pseudocode. But in a task that has a complex solution there are many aspects to the refinement. The use of repeated simulations, from different perspectives, allows the designer to attend to different aspects of the refinement in a systematic way. This allows the designer to bring his understanding to the required degree of specificity without overloading working memory.

Sketch of the Mail Designer-Soar System

In the context of Soar, problem solving is characterized as movement through successive states of knowledge in order to achieve a goal. The states of knowledge contain representations of the problem at various points in the problem-solving process. Additionally, the different aspects of the problem are regarded as different problem spaces in which different, appropriate, kinds of knowledge are brought to bear.

In order to model the design process within the Soar framework we need to provide accounts of: 1. The initial and goal state representations which form the system's input and output: In the initial state there is a description of the desired behavior of the mailer. The goal state would be a pseudocode representation of the mailer's commands. 2. The problem spaces with their appropriate operators (Table 2): The problem solving consists of trying first to apply existing knowledge relevant to a *pseudocode* problem space in order to represent the design as pieces of pseudocode. If existing knowledge is not sufficient to directly represent the design in terms of pseudocode the behavior of the mailer is simulated in *mail* space. The simulations are repeated, from varying perspectives and in increasing detail until the designer understands the system's behavior at a level that allows it to be expressed as pseudocode. The set of perspectives used in the simulations in pseudocode and mail space are generated by the strategic knowledge in the *design* problem space. Table 2 lists the operators that apply in the "design", the "pseudocode" and the "mail" problem

spaces.

Conclusions

We have described modeling software design within a Soar framework. Using this framework we are able to provide accounts for:

1. The role of schemas in bringing general knowledge to bear on knowledge about a domain: In the data presented the designer uses a high-level schema in order to create an ordered set of pairs of episodes in which a variety of inter-dependent aspects of the mailer are considered and refined.

2. The role of simulation: Simulation supports the design process in two ways. It allows the comparison of current and goal states when the current state is represented as a mechanism and the goal state is represented as behavior. Additionally, simulation supports the development of a representation of the mechanism of the system being designed when the system's behaviors are simulated at a level of detail that allows the behaviors to be expressed as mechanisms.

3. The role of progressive deepening: Because a set of simulations from a variety of perspectives are needed to complete the design we find that the same set of commands is simulated repeatedly at increasing levels of refinement.

We are optimistic that a Soar framework will continue to support detailed accounts of the mechanisms that underlie cognitve problem-solving skills.

REFERENCES

Adelson, B. and Soloway, E. The role of domain expertise in software design. *IEEE:TSE*, November, 1985.

Adelson, B. and Soloway, E. A model of software design. *International Journal of Intelligent Systems*, Fall, 1986.

Card, S., Moran, T. and Newell, A. Computer text editing. *Cognitive Psychology, 12, 1.* (1980) 32-74.

de Groot, A.D. Thought and Choice in Chess. Paris: Mouton & Cie. 1965.

Golding, A. Learning to Pronounce names by taking advice. Thesis proposal, Stanford U. 1988.

Kant, E. and Newell, A. Problem solving techniques for the design of algorithms. *Information Processing and Management*. 1984. pp. 97-118.

Laird, J. Newell A. and Rosenbloom, P. Soar: An architecture for General Intelligence. CMU CS Tech Report. 1986;

Newell A., Shaw, J. and Simon, H. A. Peport on a general problem-solving program for a computer. Proceedings of the International Conference on Information Processing, UNESCO, Paris, 1960.

Newell A. and Simon, H. A. Human Problem Solving. Prentice-Hall, 1972.

Polk, T. Fourth Annual Soar Workshop. University of Michigan, Ann Arbor. January, 1988.

Steier, D. M. Proceedings of the 1987 IJCAI Conference.

Steier, D. M. Proceedings of the 1988 AAAI Conference.

Steier, D. M. and Kant, E. *IEEE:TSE*, November, 1985.

Modeling Human Syllogistic Reasoning in Soar

Thad A. Polk and Allen Newell

Department of Computer Science,
Carnegie Mellon University

Soar is an architecture for general intelligence, which has been shown to be capable of supporting a wide variety of intelligent behavior involving problem-solving, learning, designing, planning, etc. (Laird, Newell & Rosenbloom, 1987, Steier, et. al., 1987). Soar has also been put forth as a unified theory of human cognition (Newell, 1987). We provide support for this by presenting a theory of syllogistic reasoning based on Soar and some assumptions about subjects' knowledge and representation. The resulting theory (and system, Syl-Soar/S88) is plausible in its details and accounts for existing data quite well.

The Task

Syllogisms are reasoning tasks consisting of two premises and a conclusion (Figure 1, left). Each premise relates two sets of objects (x and y) in one of four ways (Figure 1, middle), and they refer to a common set (*bowlers*). A conclusion states a relation between the two sets of objects that are not common (*archers* and *canoeists*) or that no valid conclusion exists. The three terms x,y,z can occur in four different arrangements, called *figures* (Figure 1, right), producing 64 distinct syllogisms.

Premise 1: No archers are bowlers	A: All x are y	| P1 xy | P1 yx |
Premise 2: Some bowlers are canoeists	I: Some x are y	| P2 yz | P2 yz |
Conclusion: Some canoeists are not archers	E: No x are y	| P1 xy | P1 yx |
	O: Some x are not y	| P2 zy | P2 zy |

Figure 1: Syllogism task.

Syllogisms have been much studied (see Johnson-Laird 1983 for review). The essential problem has been to understand why some syllogisms are so hard while others are so easy. However, the area is also useful as a testbed for cognitive theories.

The Soar Theory of Syllogisms

The Soar architecture has the following features:

1. **Problem spaces.** All tasks, routine or difficult, are formulated as search in problem spaces. Behavior is always occurring in some problem space.
2. **Recognition memory.** All long-term knowledge is held in an associative recognition memory, realized as a production system.
3. **Decision cycle.** All available knowledge is accumulated about the acceptability and desirability of problem spaces, states and operators for the current total context, and the best alternative is chosen among those that are acceptable.
4. **Impasse driven subgoals.** Incomplete or conflicting knowledge at a decision cycle

produces an impasse. The architecture creates a subgoal to resolve the impasse. Cascaded impasses create a subgoal hierarchy.

5. **Chunking.** The experience in resolving impasses continually becomes new knowledge in recognition memory, by means of constructed productions (chunks).

6. **Annotated models.** States are represented as annotated models (to model human cognition).

Figure 2 indicates the structure of the system: the collection of problem spaces (triangles) with operators and states. Subspaces arise from impasses, usually reflecting the need to implement operators or satisfy operator preconditions. The task data structures occur in working memory and are continually viewed by the recognition memory, which contains all task-implementation and search-control knowledge. Relevant knowledge accumulates from this memory, permitting steps to be taken in the current space or, upon impasses, creating subgoals to be solved in subspaces, etc. The micromechanics are beneath the level of detail of this paper, but drive the entire system, including learning.

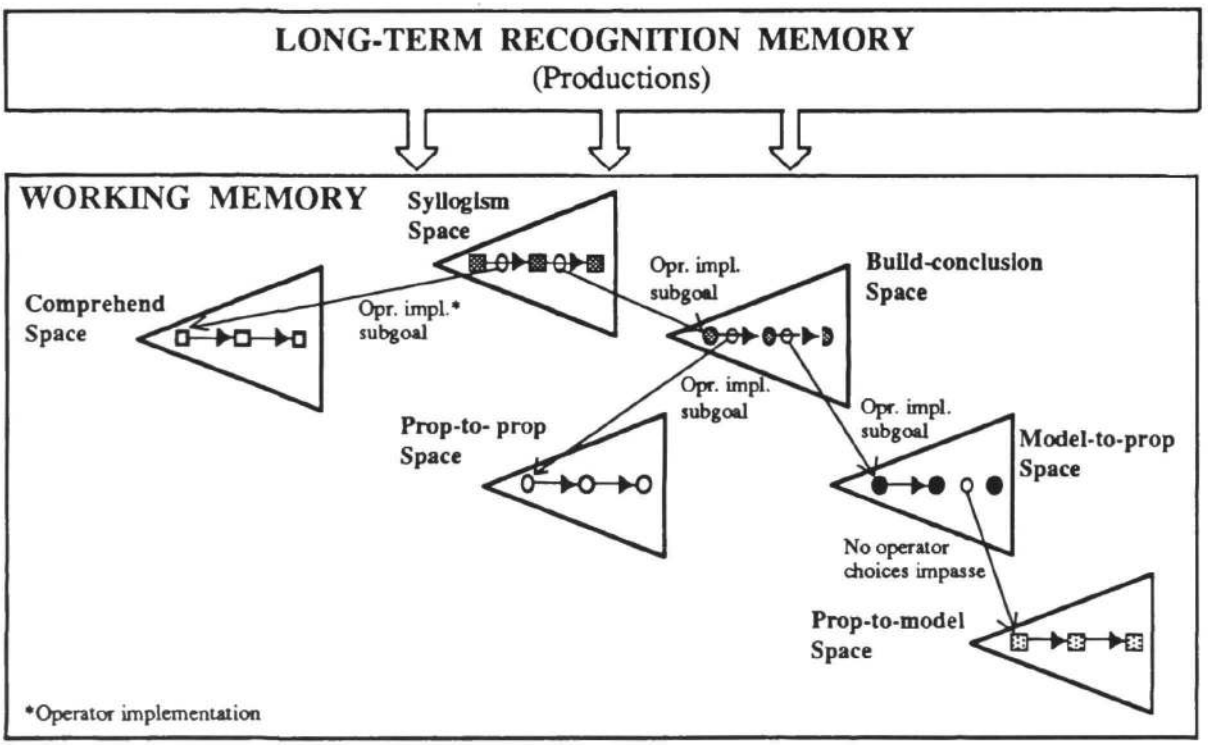
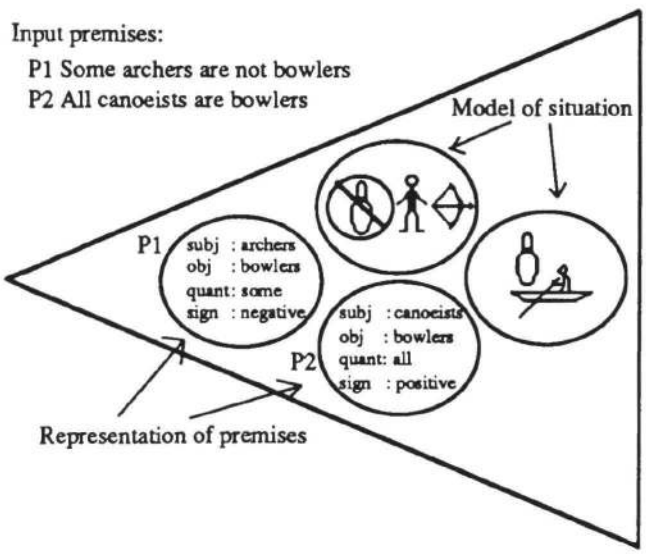


Figure 2: The structure of Soar.

A key assumption, developed strongly by Johnson-Laird (1983), is that humans represent the situations presented in syllogisms as *models*. A *pure model* is a representation that satisfies the *structure correspondence condition*: specified parts and relations of the representation data structure correspond to parts and relations of the situation, without completeness (see also Levesque, 1986). A pure model admits highly efficient *match-like* processing, but is limited in its representational power. An *annotated*

model is a representation that makes principled exceptions to a pure model, which increase its representational power, while preserving essential match-like processing. An *annotation* attaches to a data-structure part, asserting a variant interpretation for the part (e.g., **not** asserts that the part is *not* to be found in the situation where the correspondence mapping would otherwise locate it). The annotations used for syllogisms are **not**, **optional**, **many**, **target** and **source**. Annotations can quantify, but are local and do not admit unbounded processing. Figure 3 (left) indicates the models that might be built from two premises. The line through the bowling pin indicates a **not** annotation.



Syllogism Space	read-premise, build-conclusion
Comprehend Space	*all, some, no, are, not,* and one for each generic noun
Build-conclusion	generate-quantifier, generate-predicate, generate subject
Prop-to-prop	attend-to-prop, copy-subject, copy-object, copy-sign, copy-quantifier, create-auxiliary
Model-to-prop	attend-to-object, augment-proposition
Prop-to-model	attend-to-prop, augment-model

Figure 3: Annotated models, problem spaces and operators for syllogisms

Reasoning occurs by generating models to correspond to situations, inspecting the models for the properties of the situation, and forming new propositions to assert the result. Inspection is a power of the recognition memory (production match). Since models are limited, some situations can be represented only by a disjunctive set of models; reasoning then includes generating sets of models to test conjunctive properties. Reasoning with multiple models occurs in humans and has been central to model-based theories of syllogistic reasoning (Johnson-Laird, 1983, Inder, 1986), but the present theory includes only reasoning with a single model.

Six problem spaces are used in syllogistic reasoning (Figure 3 lists them, with operators, Figure 2 shows how they link together). **Comprehend**, **Syllogism** and **Build-conclusion** form the top-level path between the presented premises and the response. The knowledge to form them comes from the definition of the task, plus general skills in reading and writing. **Comprehend** is an expectation-based scheme that associates both syntactic and semantic knowledge with individual words. It constructs an initial (possibly incomplete) model; it also leaves as a byproduct a model of each premise as a proposition, with parts *subject*, *object* and *sign* (the predicate), and *quantifier*. **Prop-to-prop**, **Model-to-prop**, and **Prop-to-model** have operators required to manipulate models of situations and models of propositions, as well as attention operators to instantiate the manipulations.

The Behavior of the System

Figure 4 illustrates the system's behavior. (1) It starts in **Syllogism** and applies read-premise, implemented in **Comprehend**, to the first and then the second premise. (2) This results in an initial model, plus the two internal propositions. This encoding only extracts information about the subject of the premise. (3) Since the overall task is to produce a conclusion, build-conclusion is applied. Its space (**Build-conclusion**) puts together legal propositions. The task decomposes into discovering the subject, predicate and quantifier of the conclusion. Task knowledge permits determining some parts without other parts being specified. Incomplete or incorrect knowledge leads to composing invalid conclusions.

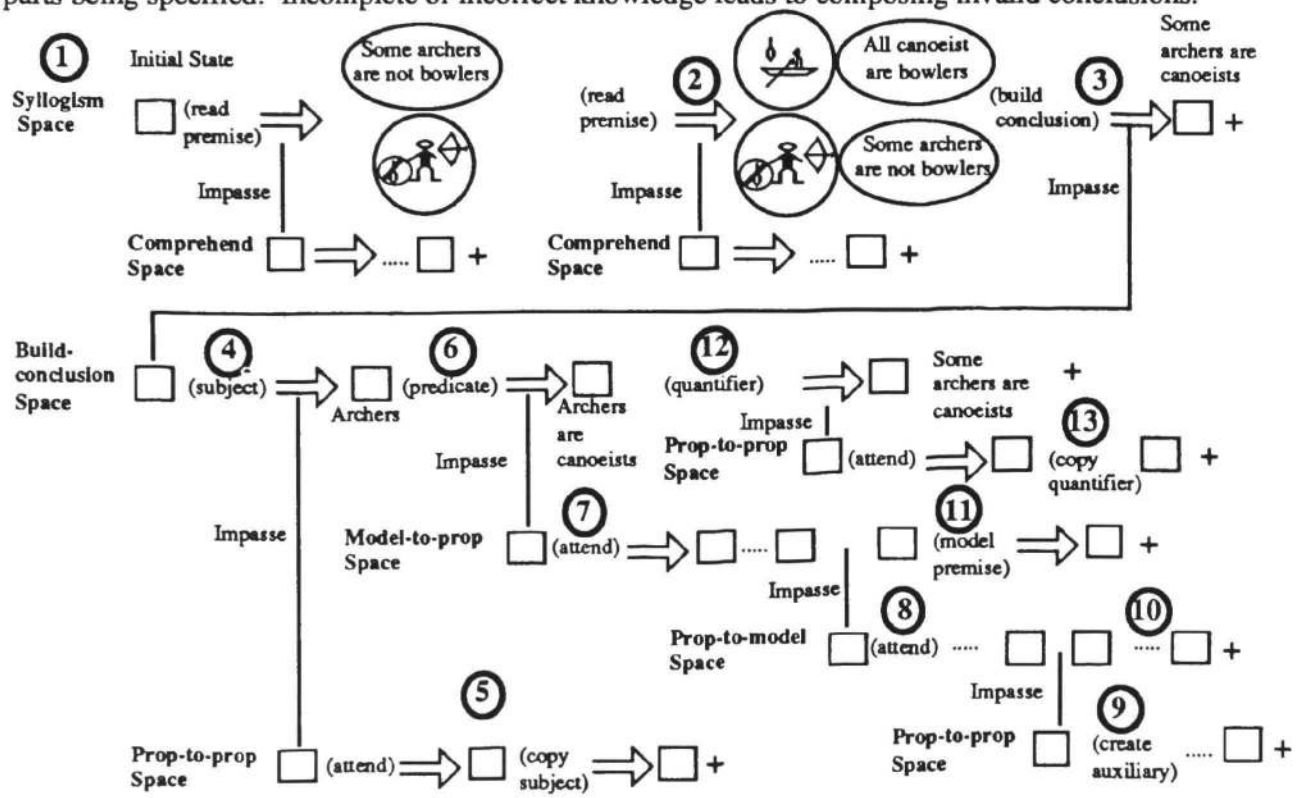


Figure 4: Behavior on *Some archers are not bowlers, All canoeists are bowlers.*

(4) Generating the subject is tried first, which uses **Prop-to-prop** because the propositions, not the model, distinguish between subjects and objects. (5) Attend-to-prop selects the first proposition and copy-subject creates the subject of the conclusion (*archers*). (6) Next, generate-predicate is selected, which uses **Model-to-prop**, because the propositions contain no useful information about the predicate. (7) The attend-to-object operator applies, but no others, because the model is incomplete. This leads to augmenting the model, using **Prop-to-model**. (8) Attend-to-prop selects premises to extract more information, but neither premise yields anything. (9) Create-auxiliary produces a new proposition in **Prop-to-prop**. It attends to the second premise and applies operators which convert it, creating the new premise *All bowlers are canoeists*. (10) This allows solving in **Prop-to-model** to resume, by focusing attention on this new proposition and using it to augment the model. (11) The model now suggests a

predicate, so solving is able to continue in **Model-to-prop** to obtain the predicate for the conclusion (*are canoeists*). (12) All that remains in **Build-conclusion** is to generate the quantifier. The model does not represent quantifiers, so **Prop-to-prop** is used again. (13) It attends to the first premise and copies its quantifier (*some*), finally obtaining, *Some archers are canoeists*. This is incorrect, but many humans fail this syllogism as well. Correctness depends on knowledge being available at many local choices.

Human Data and Soar Performance

Figure 5 presents data from (Johnson-Laird & Bara, 1984) by 20 University of Milan students on all 64 syllogisms (with unlimited time) and also the responses by Soar. The four sections of the chart correspond to the four figures (Figure 1,right). Each row corresponds to one of the 9 legal responses. The top number in each cell indicates the number of subjects giving that response to a particular syllogism. *Some archers are not bowlers* and *All canoeists are bowlers* (Figure 4) is abbreviated Oxy,Azy, and occurs in the lower left quadrant, where we see that 8 subjects responded Ixz (*Some archers are canoeists*), 7 responded Oxz (*Some archers are not canoeists*), 3 responded NVC (*no valid conclusion*) and 2 subjects gave illegal responses. Valid responses are shaded (Oxz for 7/20 correct). Only 38% percent of all responses were correct and 7 syllogisms were solved by no one.

Individual humans behave differently from each other and from themselves over time, due to learning and other factors. The data of Figure 5 are a composite, as shown by multiple responses. A family of Soar systems is required to correspond to this human variation. We varied the theory along 3 dimensions: (1) whether auxiliary propositions are created, as in our example (2 choices); (2) how premises augment objects with **not** annotations (3 choices); and (3) whether premises about *some x* augment objects about *x* (2 choices). The first dimension is one of reasoning power, the other two involve the semantics of interpreting premises. These dimensions form a family of 12 variants.

This small family accounts for 980 out of 1154 (85%) observed legal responses (126/1280 responses were illegal and not recorded) by covering 131 out of the 193 cells (68%) that contain 1 or more responses (all cells with more than 6 subjects are predicted with one exception [Oyx, Ayz = Izx]). Only one response is predicted that is not given by any subject [Oyx, Ayz = Ozx]. Frequencies were assigned to the different members of the family to produce the fit shown in parentheses in Figure 5 (15/20 subjects were assumed in the family since 23% of responses, many illegal, were unpredicted). No simple measure of fit is available, but the correlation between subjects and systems is .87.

The theory produces the classical effects, such as the *atmosphere effect* (Woodworth & Sells, 1935), the *conversion hypothesis* (Chapman & Chapman, 1959) and the *figural effect* (Johnson-Laird, 1983). Space does not permit showing the analysis, but they need only be traced out in Figure 5. The atmosphere and figural effects arise because the syntactic form of the premises serves as search control in the construction of the conclusion. The conversion effect arises when this search control is insufficient and a new proposition is created.

According to the theory, there are three main sources of difficulty: (1) making unwarranted assumptions about the premises; (2) failing to consider all the implicit ramifications of the premises; and

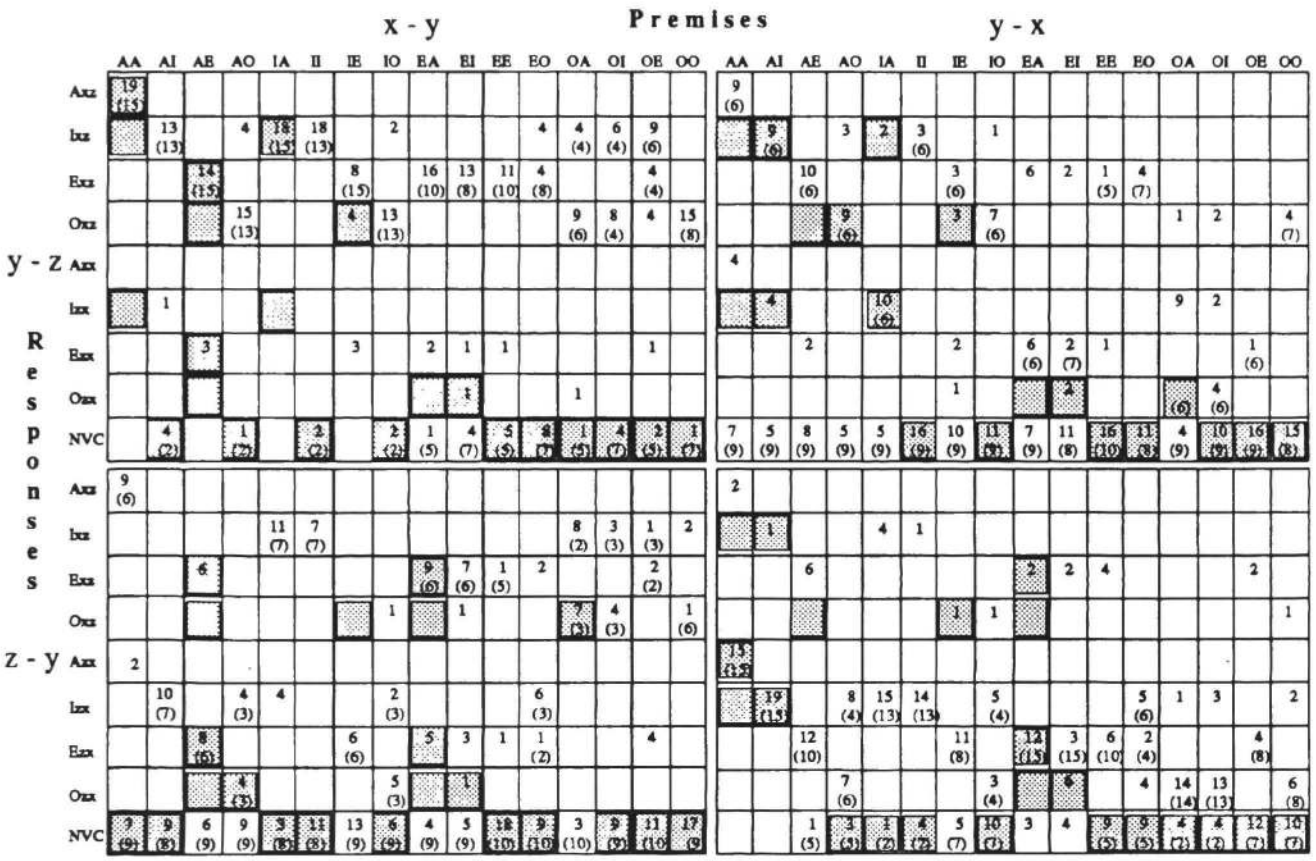


Figure 5: Data (from Johnson-Laird & Bara, 1984) and Soar predictions in ( ).

(3) failing to consider all the possible conclusions based on a (possibly correct) model. Syllogisms are difficult to the extent they present opportunities for these processing difficulties (e.g., have implicit ramifications relevant to the conclusions). This predicts that better subjects will extract more information from the premises without making unwarranted assumptions or that they will search for conclusions more extensively.

We designed a family of systems based on 10 parameters, which includes the current 3-parameter family, with the values (mostly binary) of each parameter being independently ordered by validity (so that better values correspond to more powerful and correct ways of building models). When all parameters take on their optimal values, perfect performance should occur. Better solvers should occur within this space with interpretable parameter settings. To test this, we analyzed another set of 20 subjects 58% of whose responses were correct (Johnson-Laird & Steedman, 1978). We implemented a small sub-family (24 variants including the 12) that covered 87% of the responses and 67% of the cells; it did however predict 11 responses not given by any subjects. The parameter settings of the modal system for the new distribution are better (higher in validity ordering) than those of the old distribution's modal system on 3 parameters and the same on the other 7.

The explanatory power of this theory appears better than existing theories. Their predictions are less accurate in that they predict a large number of responses that were not observed in any subjects and they do not make strong frequency predictions. Most theories only explain highly aggregate data. However, the data used here (Figure 5) is still aggregated over subjects, and nothing has yet been done with timing and protocol data. So ample opportunity remains to challenge and improve the present theory.

This theory has much to recommend it generally. It predicts flexible activity, e.g., going back to the premises to try to extract more information. Its spaces (especially executive ones) are substantially less arbitrary than prior simulations (e.g., **Comprehend** embodies a theory of elementary language comprehension). Although not reported on here, the present theory involves a theory of learning, which is an essential part of any general account of human cognitive behavior. These attributes and others arise primarily from this theory of syllogism being embedded in Soar as a unified theory of cognition.

Acknowledgements

We thank the members of the Soar project for support and criticism, especially Rick Lewis who is working on **Comprehend**; also Norma Pribadi for making the beautiful figures and Phil Johnson-Laird for comments on this theory. This research was supported by the Information Sciences Division of the Office of Naval Research under Contract N00014-86-K-0678 and also by the NSF under the Engineering Research Center Program, Contract CDR-8522616. 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

Chapman, I. J., & Chapman, J. P. (1959). Atmostphere effects re-examined. *Journal of Experimental Psychology, 58*, 220-226.

Inder, R. (1986). Modeling syllogistic reasoning using simple mental models. In Cohn, A. G., & Thomas, J. R. (Eds.), *Artificial Intelligence and its Applications*. New York: Wiley.

Johnson-Laird, P. (1983). *Mental Models*. Cambridge, MA: Harvard.

Johnson-Laird, P. N., & Bara, B. G. (1984). Syllogistic inference. *Cognition, 16*, 1-61.

Johnson-Laird, P. N., & Steedman, M. (1978). The psychology of syllogisms. *Cognitive Psychology, 10*, 64-99.

Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence, 33*, 1-64.

Levesque, H. J. (1986). Making believers out of computers. *Artificial Intelligence, 30*, 81-108.

Newell, A. (1987). Unified Theories of Cognition. The William James Lectures. Harvard University, Spring 1987. (Available in vidocassette from Harvard Psychology Department).

Steier, D. E., Laird, J. E., Newell, A., Rosenbloom, P. S., Flynn, R. A., Golding, A., Polk, T. A., Shivers, O. G., Unruh, A. & Yost, G. R. (1987). Varieties of Learning in Soar: 1987. In *Proceedings of the Fourth International Workshop on Machine Learning*. Los Altos, CA: Morgan Kaufman.

Woodworth, R. S., & Sells, S. B. (1935). An atmosphere effect in formal syllogistic reasoning. *Journal of Experimental Psychology, 18*, 451-460.

INTEGRATED COMMONSENSE AND THEORETICAL MENTAL MODELS IN PHYSICS PROBLEM SOLVING

Jeremy Roschelle, Institute for Research on Learning
3333 Coyote Hill Road, Palo Alto, CA 94304
&
Education: Math, Science, and Technology Program
University of California, Berkeley

The whole of science is nothing more than a refinement of everyday thinking. It is for this reason that the critical thinking of the physicist cannot possibly be restricted to the examination of concepts of his own specific field. He cannot proceed without considering critically a much more difficult problem, the problem of analyzing the nature of everyday thinking.

Albert Einstein, quoted in Miller (1986)

Abstract and Introduction

Cognitive Scientists have recently developed models of physicists' problem solving behavior. Their models propose a rich set of cognitive constructs including procedures (Heller and Reif, 1984), problem-solving schemata (Larkin 1983), categorization rules (Chi, Feltovich & Glaser, 1981), phenomenological primitives (diSessa 1983), forward and backward chaining (Larkin, McDermott, Simon, & Simon, 1980), and qualitative reasoning (deKleer, 1975, Forbus 1986, deKleer and Brown, 1986, and others in Bobrow, ed. 1986). These constructs have proved useful in understanding aspects of physics reasoning.

This paper will provide an analysis of physics problem solving skill that integrates cognitive constructs previously considered disparate. The main point is this: *Commonsense reasoning about situations provides an indispensable resource for coping with physics problem solving complexity.* More precisely, I will argue that the systematic integration of the deep structure of situational and theoretical knowledge can reproduce competent physics cognition. To support this claim I will discuss the capabilities of running computer programs, written in Prolog, that implement several representations and reasoning processes. In addition, I will show how the Prolog models capture the essence of a think-aloud protocol of a physicist recovering from an error while working a novel problem.

The Problem Domain

This research concerns a domain of problems like those found in physics textbooks. (See figure 1 for examples.) In these problems, blocks can be connected by strings, and can touch fixed surfaces. All blocks are assumed to have zero initial velocity. Four kinds of forces appear in this domain, gravity, tension, normal forces and "given" forces. In each problem the goal is to find the unknown accelerations, tensions, and normal forces. A generative grammar (table 1) can produce an infinite supply of problems in this domain.

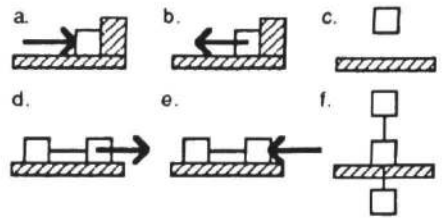


Figure 1: Problems in the Domain

Table 1: Problem Domain Grammar

Problem <- SituatedBlock
Problem <- SituatedBlock, string, Problem
SituatedBlock <- ForcedBlock
SituatedBlock <- surface, ForcedBlock
SituatedBlock <- ForcedBlock, surface
ForcedBlock <- block
ForcedBlock <- force, block
ForcedBlock <- block, force

This domain is interesting because of the difficulties it poses for the theorist. Two levels of complexity in the domain lead to two criteria that a successful theory should meet. The primary complexity resides in the mapping from physical situations to scientific models. Objects like strings and walls do not have a simple representation in Newtonian physics; a first-principles explanation for their behavior can only be expressed in terms of Quantum Mechanics. Physicists, however, approximate interactions involving strings and walls with Newtonian models in order to expedite the solution process. A proposed theory should account for physicists' abilities to reliably generate approximate Newtonian models for observable physical situations.

A secondary complexity arises in the process of manipulating mathematical representations — solving large sets of equations is hard. Consider figure 1f. Eight unknown variables appear in this physical situation, potentially requiring the solution of eight simultaneous linear equations. Yet most physicists could determine all the unknowns precisely while solving only a single equation (T=mg for the hanging block). A physicist might explain this situation by saying something like this:

"I can see that the middle block will be supported by the table, the block above the table will fall, and the block below the table will hang on the string. The acceleration of a supported block or a hanging block is zero, while the acceleration of a falling block is a known constant, $g = -9.8\ m/s^2$. The tension in the top string will be zero because it is collapsing under the falling block. The tension in the bottom string will be enough to balance out gravity. This force can be computed by multiplying the mass, m, and the gravitational constant, g."

Note the prevalent use of concepts like supporting, falling, hanging, collapsing, and balancing in this explanation. diSessa (1983) has argued that concepts like these, called phenomenological primitives (p-prims), provide the deep structure for intuitive physical reasoning. While these concepts are have no formal role in Newtonian science, I will show that they can streamline the solution process.

Representational Framework

The complexities of situation-to-theory mapping and of solving large sets of linear equations together point to the need for a representational framework that integrates multiple representations. I propose the framework illustrated in figure 2, called the "Relational Framework." The Relational Framework comprises a *situational representation* and a *theoretical representation*. This framework builds on representational distinctions developed in McDermott & Larkin (1978) and the use of qualitative reasoning pioneered by deKleer (1975), with two crucial additions:

1. Both representations are mental models. (Johnson-Laird , Holland, et. al., 1896, Gentner & Stevens, 1983)
2. Both mental models can use qualitative reasoning. (Forbus 1986, deKleer and Brown, 1986, and others in Bobrow, ed. 1986)

The situational representation contains the kinds of objects, properties, and relations typically found in real world situations. For example, it might include objects like blocks, tables, and strings; properties like heaviness, roughness, and springyness; and relations like on-top-of, next-to, and touching. P–prims allow for structured explanation of behavior in the situational mental model. Qualitative reasoning allows the behavior in situational mental models to be simulated, generating expectations for future behavior.

The theoretical representation contains the kinds of objects, properties, and relations found in a scientific theory. Since this paper is primarily concerned with simple classical mechanics, the theoretical representation will include point masses, momentum, and forces. The conceptual structure of the theoretical model derives directly from Newtonian Mechanics. Qualitative Reasoning can generate predictions about the behavior of a theoretical mental model using the process of envisioning. (deKleer and Brown, 1986)

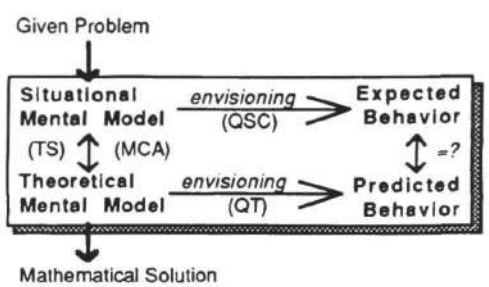


Figure 2: The Relational Framework

The following sections compare the strengths and weaknesses of four models of physics reasoning, each which implements part of the relational framework. The first, the Textbook Solution model (TS) is simple to compute, but lacks the search capability necessary to build the situation-to-theory mapping for all problems in the domain. The remaining three models all use a case analysis search procedure. The Mathematical Case Analysis Model (MCA) can identify a correct theoretical description of a given situation, but leads to an explosion in computational complexity. The Qualitative Theoretical Case Analysis Model (QT) reduces this computational complexity via qualitative reasoning, but often gets trapped in ambiguities. The Qualitative Situational Causal Model (QSC) also reduces the complexity of case analysis via qualitative reasoning, but uses a situational rather than a theoretical representation. While QSC gets trapped in ambiguities less frequently, its knowledge source may contain misconceptions.

These models are presented as competitors to highlight their unique characteristics. However to represent human cognition, several models might be deployed in parallel, as indicated by the parallel envisioning paths in figure 2. A problem solver could exploit the redundancy of parallel models in order to find inconsistencies and trap errors. Alternatively, a problem solver could increase productivity by replacing some mathematical computations with qualitative reasoning. A protocol segment will later illustrate how one physicist integrated Qualitative Situational and Qualitative Theoretical reasoning processes to achieve an efficient and error-free solution.

Textbook Solution Procedures

The Textbook Solution (TS) model allows an examination of the sufficiency of standard textbook problem solving procedures for this domain. Standard textbook procedures *should* be sufficient for this domain, since string tensions and normal forces are part of the standard curriculum.

Textbook procedures share several characteristics. First, these procedures use a series of representations and transition rules. Larkin and McDermott (1978) identify this series as having four representations, words, a situation sketch, a theoretical sketch, and mathematical symbols. Second, textbook procedures examine only the surface features of a situation sketch. Sample surface features in a sketch would be "the blocks are touching" "a block is on a table," and "a string is attached to a block." Third, textbook procedures do not invoke backtracking or retraction of previously derived information. The four kinds of representations, from words to equations, follow a forward progression, in which each later representation borrows from the earlier one, but does not modify it. The exclusion of backtracking and retraction severely limits problem solving capability in this domain.

The Prolog TS model follows the steps below, adapted from Kleppner and Kolenkow's introductory physics text (1973):

1. Identify systems that can be treated as particles.
2. Identify all forces present.
3. Write an instance of Newton's Second Law in the vertical and horizontal directions for each particle.
4. Write an instance of Newton's Third Law for each equal and opposite force pair.
5. Write additional constraints as necessary.
6. Solve the equations, by keeping track of known and unknown variables.

Steps two and four are accomplished using Heller and Reif's (1984) procedure. The TS model follows the first five steps and outputs a set of n equations in n unknowns. These equations could presumably be solved by a computer using an algebraic algorithm. In this case, the author solved them by hand.

The TS model can solve some simple physics problems. However, a large class of simple physics problems exists which the TS model cannot solve. This class of problems includes many problems involving the normal force and string tensions since these forces can only be approximately represented in Newtonian Mechanics. Finding the correct approximate representation requires a search with backtracking capabilities, not provided for in the TS Model.

The TS model, for example, cannot solve the problem in figure 1f because it does not have enough information. (See table 2 for a step-by-step application of the TS model to figure 1f). The missing information is that the top block is falling, that the other two blocks are stationary, and that the top string has no tension. This information could be inferred by a human problem solver using commonsense, however without appropriately integrated commonsense knowledge, the TS model cannot find the correct solution. One way of integrating commonsense will be discussed later in the QSC model. But first, two models that avoid the need for representing commonsense knowledge are introduced.

Table 2: An Application of the TS Model to Figure 1f

1. The three blocks can be treated as particles in this situation.
2. The top block has two forces, the string tension, T_1 and gravity, m_1g. The middle block has four forces, the string tension T_2, the string tension T_3, gravity, m_2g and the normal force, N. The bottom block has two forces, the string tension, T_4, and gravity, m_3g.
3. $-T_1 + m_1g = m_1a_1$
   $T_2 - T_3 + m_2g + N = m_2a_2$
   $T_4 + m_3g = m_3a_1$
4. $T_1 = T_2$; $T_3 = T_4$
5. No additional constraints given.
6. Only 5 equations in 8 unknowns, no solution possible.

Mathematical Case Analysis

As mentioned above, the root cause of the problems with the TS model is that tension and normal forces do not have a simple Newtonian model. Newtonian Mechanics, however, can make predictions in situations involving complex force functions by invoking case analysis.

Case analysis is a procedure commonly taught in engineering disciplines by which a complicated function, like the normal force, is divided into several distinct operating regions, each which can be represented by a simple function. The tension force for an ideal string, for example, can be separated into four operating regions, mainly:

1. string collapsed; distance < length, Tension = 0
2. string collapsing; distance = length, Δdistance < 1, Tension = 0
3. string taut; distance = length, Δdistance = 0, Tension > 0
4. string breaking: distance = length, Δdistance > 0, Tension = maximum load capacity of the string

(Note: The models discussed here actually use a simpler breakdown for the ideal string which ignores the collapsed and breaking states.)

A Prolog model called MCA (Mathematical Case Analysis) uses such decompositions to implement case analysis as a search for a consistent set of linear equations. MCA builds and examines a search tree that represents each possible combination of operating regions for the tensions and normal forces that exist in the system. At each leaf in the tree, MCA generates a representation that assumes each local part of the system is operating in certain regions (or cases). If this representation is globally consistent, then it is a solution. Otherwise that particular combination of operating regions can be eliminated.

To build a scientific representation, the MCA model follows the first four steps of the TS model. In step 5, the MCA model adds a non-deterministic choice of operating region for each normal force and string tension. It outputs a set of linear equations, which are currently solved by hand in step 6. If the equations are inconsistent, MCA backtracks to step 5 and chooses another set of operating regions. Thus if there are n complex functions requiring m operating regions each, MCA searches a tree with m^n leaves.

Applied to figure 1f, MCA produces the same equations as TS for steps 1-4. In step 5, MCA chooses one of 8 possible sets of operating regions. There are eight sets of operating regions because there are two strings and one normal force, each requiring a breakdown into 2 operating regions, and 2^3 is 8. The resulting additional equations for two of the eight sets of operating regions is shown below in table 3. The first, when combined with the other equations, is inconsistent and must be rejected. The second is consistent, and therefore is a solution.

Table 3: The MCA model, 2 search points for figure 1f

Operating Regions Set A: (inconsistent)

$T_1 > 0, a_1 = a_2;$

$N > 0, a_2 = 0;$

$T_3 > 0, a_2 = a_3$

Operating Regions Set B: (consistent)

$T_1 = 0, a_1 < a_2;$

$N > 0, a_2 = 0;$

$T_3 > 0, a_2 = a_3$

Table 4: The QT model, 2 search points for figure 1f

some equations with operating region A	qualitative form
$-T_1 + m_1g = m_1a_1$	$-1 + -1 = a_1$ #
$a_1 = a_2$	$a_1 = a_2 = 0$ #
$T_2 - T_3 + m_2g + N = m_2a_2$	$1 - 1 + -1 + 1 = a_2$ *
$T_4 + m_3g = m_1a_3$	$1 - 1 = a_1$*

some equations with operating region B	qualitative form
$T_2 - T_3 + m_2g + N = m_2a_2$	$0 - 1 + -1 + 1 = a_2$ *
$T_4 + m_3g = m_1a_3$	$1 - 1 = a_1$*

(# means inconsistent, * means ambiguous)

While this procedure can solve all problems in the domain, it involves solving 8 sets of 8 equations in 8 unknowns. This is a lot of work, even for a computer. Moreover, the MCA model does not support a very satisfying explanation of the solution — the explanation, essentially, is that the computer followed the case analysis procedure and identified a consistent set of equations, which must be the solution because the case analysis procedure is correct. There is no evidence that physicists would use a procedure like MCA to solve this problem.

Qualitative Reasoning, Scientific Representation

The primary virtue of case analysis is that it can bring a situation that includes tension and normal forces into the range of applicability of Newtonian Mechanics. The drawback to case analysis is combinatorial explosion in the number of sets of simultaneous linear equations to be solved. Qualitative Reasoning cannot reduce the combinatorial explosion, however it can reduce the effort involved in checking the consistency of each set of equations.

To apply Qualitative Reasoning to this domain, I follow Reif and Heller's (1982) suggestion to check the consistency of the direction of acceleration predicted by the resultant force with constraints on acceleration. The first step is to make each quantitative $\Sigma F = ma$ equation into a qualitative one. To do this, I replace each quantitative force variable with the number 1 if its sign is positive, and the number -1 if its sign is negative. The left side of the equation is then summed according to the qualitative arithmetic table (Forbus, 1986), yielding a predicted sign for the acceleration. This can be compared with any constraints on the sign of the acceleration.

The QT Prolog model carries these steps out computationally. The first five steps are the same as in the MCA model, however in step six the equations are converted to qualitative form and checked for qualitative consistency. Table 4 shows the result of the QT model applied to figure 1f. The equations with a hash mark (#) are inconsistent, while the equations with an asterisk (*) are ambiguous. The QT model shows operating region A to be inconsistent. However operating region B, found consistent by MCA, cannot be proved consistent by QT because it is qualitatively ambiguous.

The QT model has two major advantages over the MCA model. First, the QT model can identify inconsistent sets of equations without extensive algebraic manipulation — it needs only to compute the sign of acceleration from each instance of Newton's Second Law and compare the result to constraints on the acceleration. The QT model therefore reduces the amount of computation needed to solve a physics problem in this domain. Second, the QT model lends itself to understandable explanations. For example, one could explain the contradiction in operating set A in table 4 as follows:

"There are two forces on the topmost block, gravity and the tension on the string. Since both act downwards, the block will accelerate downwards. However, there can only be tension in the string if the blocks at either end are moving at the same rate. Since the block on the table is not accelerating and the block above it is, the blocks at either end of the string are not accelerating at the same rate. The assumption that there is tension in the string therefore leads to an inconsistent prediction."

The disadvantage of the QT model is that it often yields ambiguous results, as is the case with operating region b in table 4. While the next model, the QSC model can also create ambiguities, it does so less frequently.

Qualitative Reasoning, Situational Representation

All three previous Prolog models share two important characteristics: (1) they operate primarily on a problem representation based on scientific entities like force and mass, and (2) the reasoning in the models is constraint-driven, rather than causal. This section presents the Qualitative Situational Causal (QSC) model. As its name suggests, QSC reasons causally about a situational model. As the Prolog model demonstrates, QSC can make certain crucial inferences more efficiently and effectively than the reasoning components discussed previously.

The QSC model distinguishes between two classes of interactions, *tendency-producing interactions* and *constraint-producing interactions*. (These terms are introduced to avoid confusion between commonsense and scientific use of the word "force.") Tendency-

producing interactions (t-interactions) are the ultimate causes of motion. T-interactions have a value that is independent of state of other interactions in the situation. Gravity and given forces are the t-interactions in the present domain.

Constraint-producing interactions (c-interactions), on the other hand, are conditional of the presence of other interactions. The two c-interactions in this domain are string tension and the normal force. These interactions respond to applied forces so as to maintain some state of affairs (a constraint). Strings react to applied forces so as to keep to connected objects at a particular separation. The normal force reacts to applied forces so as to keep one surface from passing through another.

As with the MCA and QT models, the QSC model is based on case analysis. (See table 5.) However, unlike these models the QSC model does not first perform steps 1-4 of the textbook solution model, because the QSC model operates with the situation representation directly.

Like the previous models, QSC non-deterministically chooses and evaluates a set of operating regions, using one choice of operating region for each c-interaction. To evaluate a set of operating regions, QSC checks each c-interaction for local consistency, assuming the other c-interactions fixed.

Each consistency check requires evaluating a statement like "The string is loose, and if the string were not there the blocks on either end of it would not move apart." To evaluate this statement, the QSC model must calculate the motion of the blocks on either end of the string. The QSC model makes the motion calculation qualitatively, and causally. The calculation is qualitative because the QSC model computes only the sign of the acceleration (by using the qualitative arithmetic table to sum t-interactions). The calculation is causal because only t-interactions are included. To eliminate c-interactions from the calculation, propagation rules are applied. These rules in effect say that t-interactions will propagate through taut strings, but not through resisting walls.

Like the MCA and QT models, the QSC model might have to search 8 sets of operating regions to solve figure 1f. Table 6 shows an English translation of the behavior of the model on this problem. Only two sets of operating regions are shown, corresponding to the operating regions chosen in previous examples.

Table 5: C-Interactions For Strings and Surfaces

Surface c-interaction

case a: There is pressure on the block from the surface, and if the surface were not there the block would move through the space occupied by the surface

case b: There is no pressure on the block from the surface, and if the surface were not there the block would not move through the space occupied by the surface.

String c-interaction

case a: The string is taut, and if the string were not there the blocks on either end of the string would move apart.

case b: The string is loose, and if the string were not there the blocks would not move apart.

Table 6: The QSC Model

Operating region set a:

both strings case a, surface case a

1. For the top string, if the string were not there would the blocks move apart? Assuming the string not there, there is only one t-interaction acting on the top block, gravity. The top block will therefore move downwards. The middle block is touching a resisting surface, therefore it is not moving. The blocks are moving closer together, so the answer is no, and this set of operating regions is inconsistent. The other c-interactions do not need to be evaluated.

Operating region set b: top string case b, bottom string case a, surface case a.

1. If the top string were not there, would the blocks not move apart? As in number 1 above, the top block would move downwards, so the answer to this question is yes.
2. If the bottom string were not there, would the blocks move apart? There is only one t-interaction acting on the bottom block, the force of gravity. The bottom block will there move downwards. As in number 1, the middle block will not move. Since the blocks are moving apart the answer is yes.
3. If the surface were not there, would the block move through the space it occupies? There are two t-interactions acting on the middle block, the force of gravity acting on it, and the force of gravity acting on the bottom block, propagated through the taut string. Both are downward so the block would move downward through the space occupied by the surface. The answer is yes.

Since the answer to all three questions is yes, this set of operating regions is consistent.

As table 6 shows, the QSC model can positively identify both inconsistent and consistent operating regions for this problem. Once a set of consistent operating regions is identified, the QSC model can proceed with the textbook solution (TS) model. In step 5, the operating regions discovered by the QSC model guide the addition of suitable constraints. The equations can then be solved for a mathematical solution if desired.

The QSC model fails only when two t-interactions act on the same body, and even then only under certain circumstances. When this failure occurs, it amounts to a failure of the qualitative logic because of an ambiguity. These ambiguities occur less frequently in the QSC model than the QT model, because the QSC model effectively eliminates all c-interactions, while the QT model represents each c-interaction as a force.

In addition, QSC models can fail in the presence of misconceptions. However, diSessa (working paper) has suggested that physicists adjust their p-prims through learning so that the p-prims more accurately reflect causal processes in the world. Thus physicists could apply p-prims without necessarily invoking misconceptions. Moreover, when physicists' p-prim representation is integrated with their theoretical representation, physicists can gain the efficiency of qualitative situational representations, without losing the robustness of theoretical representations.

Protocol Analysis

The models above will now be applied to the analysis of a physicist's think-aloud protocol. This protocol resulted from an interview with a University of California, Berkeley physics graduate student with significant teaching experience. In the interview, the subject was shown the sketch in figure 3, and asked "what's happening." The transcript of his response appears in table 7.

Roschelle and Greeno (1987) includes an extensive analysis of this protocol, the highlights of which are summarized here. In step one, the subject quickly comes to a narrow focus of attention, which is remarkable given that 13 forces would appear in a theoretical representation of the situation. Instead, we conjecture that the subject uses a p-prims and a situational representation. This hypothesis is supported is step 2, when the subject envisions the motion of the situational model. In step 3 the subject builds a theoretical model of the situation, and envisions it. In step 4 the subject recognizes that the results of situational and theoretical envisioning conflict. Finally, in step 5 the subject finds a bug , "friction opposes motion" in his theoretical model and replaces it with a the correct rule, "friction opposes relative motion."

This protocol is an excellent example of the relational framework, because it shows the continuous interplay between situational and theoretical mental models, as well as the role of situational deep structure in coming to a quick focus of attention. It can be modeled quite effectively with elaborations of the QSC, TS and QT models.

Table 7: One physicist's protocol for figure 3

1. This one is very interesting because it illustrates a very important point which is that friction isn't always in the wrong direction -- or the right direction. The thing that bothered me right away is the extra mass sitting here [on top]. And the question is: what's going to happen with the extra mass.
2. Well if there's everything accelerating to the right ,then so will this top mass.
3. And on the other hand , your first temptation is to draw a force diagram in your head and you say 'OK the thing's moving to the right, so friction is to the left.' Except that friction is the only force moving it —
4. — so right away you reach a problem.
5. The answer is ... that friction opposes relative motion, so the friction's going to go in whichever direction it needs to point to oppose the motion between the little mass and the big mass it's sitting on. And that happens to be to the right if the big mass is moving to the right.

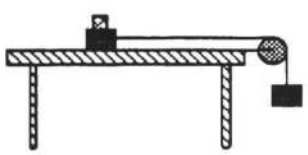


Figure 3: Extra Block Sketch

The QSC model already contains situational reasoning capability for blocks, strings, and supporting surfaces. Two additional rules would be needed to produce the prediction of motion to the right (protocol segment #2). First, a rule is needed to handle the c-interaction of a pulley. This rule would be analogous to the rule that propagates t-interactions through strings. The only difference is that the pulley c-interaction changes the direction of the t-interaction as it goes over the pulley.

A second rule is needed to handle the c-interactions between two blocks that can slide relative to each other. This rule would propagate t-interactions that are parallel to the sliding surfaces through the surface-to-surface contact. This rule models the behavior of friction in transmitting force.

Using these two rules, QSC will predict that the extra block moves to the right, because the only t-interaction on the extra block is the gravitational interaction that has been propagated though the pulley system and the surface-to-surface contact.

In order to generate a theoretical representation, the TS model needs rule for identifying and representing friction. There are two forms of this rule in the protocol, "friction opposes motion" and "friction opposes relative motion." To model the protocol, first one rule and then the other is added to the TS model.

The QT model already has the capability to predict the direction of acceleration based on a theoretical representation. If the friction on the extra block is to the left, QT will predict motion to the left, as in the protocol.

One additional relational component necessary to model this protocol is a procedure that compares the predicted behavior of the QSC and QT models. In this case, the procedure would compare the direction of motion in the situational representation with the direction of acceleration in the theoretical representation. With the buggy friction rule in effect, this comparison will fail. With the correct friction rule in place, this rule will succeed. Table 8 shows the sequence of events corresponding to the protocol.

Notice that the physicist uses multiple representations in two ways in this problem. In protocol line #2, he uses QSC reasoning to avoid the work of determining the motion of the pulley system via a scientific representation. Later, in protocol line #4, he uses situational reasoning in parallel with scientific reasoning to identify an error. It is especially interesting that in this case the physicist's situational representation was correct while his initial scientific representation was wrong!

Table 8:Computational Model of the Protocol

1. The QSC model predicts motion to the right.
2. The TS model builds a theoretical model using the rule "friction opposes motion."
3. The QT model predicts acceleration to the left.
4. The comparison procedure detects a conflict.
5. The programmer, acting in the place of a complex retrieval process, replaces the buggy friction rule with the correct one.
6. The TS model re-builds the theoretical model.
7. The QT model prediction acceleration to the right.
8. The comparison procedures finds the models to be consistent.

Conclusion

Discussions of p-prims and situational reasoning have generally been restricted to the question, "what's wrong with novices?" The QSC model suggests another question to which p-prims and situational reasoning might be the answer, mainly *"what's right about experts?"* In particular, the Prolog models discussed above have shown that the integration of situational and theoretical deep structure can result in performance that is both efficient and robust. The protocol example is a strong example of efficient and robust problem solving; while the subject analyzes the situation quickly, he is simultaneously able to detect and correct a bug in his theoretical model. The development of parallel situational and theoretical representations, as well as the use of qualitative reasoning, makes expert competence possible without sacrificing expert performance.

Acknowledgements

Thanks to Jim Greeno, Andy diSessa, Peter Pirolli, and Alan Schoenfeld for enthusiasm, ideas, and scaffolding. Thanks also to Susan Newman and others at the Institute for Research on Learning for reading early drafts of this paper.

The author may also be contacted by e-mail addressed to "jeremy@soe.berkeley.edu"

References

Bobrow, D.G., Ed. (1986), Qualitative Reasoning about Physical Systems, Cambridge, MA ,MIT Press.

Chi, M. T. H., Feltovich, P. J., & Glaser, R. (1981), Categorization and representation of physics problems by experts and novices. Cognitive Science, **5**, 87-119.

deKleer, J. (1975), *Qualitative and Quantitative Knowledge in Classical Mechanics*,MIT Master's Thesis.

deKleer, J. and Brown, J.S., 1986, *A Qualitative Physics Based on Confluences*, in D.G. Bobrow (Ed.) Qualitative Reasoning about Physical Systems, Cambridge, MA MIT Press.

diSessa, A.A. (1983), *Phenomenology and the Evolution of Intuition*, in Gentner, D. and Stevens, A. (eds.), Mental Models, Hillsdale, NJ, Lawrence Earlbaum Press.

diSessa, A.A., (working paper), *Towards an Epistemology of Physics*.

Forbus (1986), *Qualitative Process Theory*, in D.G. Bobrow (Ed.) Qualitative Reasoning about Physical Systems, Cambridge, MA MIT Press.

Heller, J.I., and Reif, F (1984), *Prescribing Effective Human Problem Solving Processes: Problem Description in Physics*, Cognition and Instruction, **1**, p. 177-216.

Holland, Holyoak, Nisbett, & Thagard (1986), Induction, Cambridge, MA, MIT Press.

Johnson-Laird, P.N. (1983), Mental Models, Cambridge, MA, Harvard University Press.

Kleppner, D. and Kolenkow, R.J. (1973) An Introduction to Mechanics, San Francisco, McGraw-Hill.

Larkin, J.H. (1983), *The Role of Problem Representation in Physics*, in Gentner and Stevens, Mental Models, Hillsdale, NJ, Lawrence Earlbaum.

Larkin, J.H., McDermott, J., Simon, D.P., and Simon, H. (1980) *Models of Competence in Solving Physics Problems*, Cognitive Science, **4**.

McDermott, J. & Larkin, J.H. (1978) *Re-representing textbook physics problems*, The Canadian society for computational studies of intelligence, University of Toronto Press.

Miller, A.I. (1986), Imagery and Scientific Thought, Cambridge, MA, MIT Press.

Reif, F., and Heller, J.I. (1982) *Knowledge Structure and Problem Solcing in Physics*, Educational Psychologist, **17**, p. 102-127.

Roschelle, J. and Greeno, J. (1987), *Mental Models in Expert Physics Problem Solving*, ONR Report GK-2, available from University of California, Berkeley, School of Education.

A connectionist model of selective attention in visual perception

Michael C. Mozer

Institute of Cognitive Science
University of Colorado, Boulder

This paper describes a model of selective attention that is part of a connectionist object recognition system called MORSEL. MORSEL is capable of identifying multiple objects presented simultaneously on its "retina," but because of capacity limitations, MORSEL requires attention to prevent it from trying to do too much at once. Attentional selection is performed by a network of simple computing units that constructs a variable-diameter "spotlight" on the retina, allowing sensory information within the spotlight to be preferentially processed. Simulations of the model demonstrate that attention is more critical for less familiar items and that attention can be used to reduce inter-item crosstalk. The model suggests four distinct roles of attention in visual information processing, as well as a novel view of attentional selection that has characteristics of both early and late selection theories.

Few would argue that the visual system is unlimited in its capacity for processing sensory information. Some means of selective and sequential analysis is required. This is the primary function of attention: to control the amount and the temporal order of information flowing through the visual system. Any complete model of visual information processing must thus address the issue of attention. In this paper, I describe an attentional mechanism designed for a connectionist model of two-dimensional object recognition called *MORSEL* (Mozer, 1987a, b). MORSEL is capable of identifying multiple objects presented simultaneously on its "retina," but because of capacity limitations, MORSEL requires an attentional mechanism to prevent it from trying to do too much at once and making errors.

Briefly, MORSEL (Figure 1) consists of four components: (1) a set of processing modules that analyze objects along various attribute dimensions; (2) a network that constructs a consistent interpretation of the perceptual data provided by these modules (the *pull-out net*); (3) an *attentional mechanism* (*AM* for short) that guides the efforts of the modules; and (4) a *visual short-term memory* that holds object descriptions. To illustrate the typical operation of the system, consider a simple example in which MORSEL is shown a display containing two colored letters, a red X and a blue T. These letters will cause a pattern of activity on MORSEL's retina, which serves as input to each of the processing modules as well as to the AM. The AM then focuses on one retinal region, say the location of the red X. Information from that region is processed by each module. One module extracts shape information, identifying the object as an "x" or possibly a "y," another extracts color information, identifying the object as being red. The pull-out net then selects the most plausible interpretation of each module's output, in this case "x" and "red." The representation at this level of the system encodes attributes of the visual object *without regard to location.* Location information is recovered from the AM, which indicates the current location of focus. Shape, color, and location information are then bound together and stored in the short-term memory. Next, attention shifts to the blue T, and this process repeats.

I have built a computer simulation of MORSEL with one module elaborated in detail — a letter and word recognition system called *BLIRNET*. BLIRNET has been trained to recognize letters and words in arbitrary retinal locations, and is able to recognize several items simultaneously, although interactions within the network limit the number of items that can be accurately processed. BLIRNET is a hierarchical multi-layered network. Its input layer is a retinotopic feature map arranged in a 36×6 spatial array, with detectors for five feature types at each point in the array (line segments at four orientations and line-segment terminator detectors). Letters of the alphabet are encoded as an activity pattern over a 3×3 retinal region. BLIRNET's output layer contains *letter-cluster detectors,* which respond to single

My thanks to Don Norman, Hal Pashler, and Geoff Hinton for their guidance, and to Steve Nowlan for helpful comments on an earlier draft. This work was supported by grant 87-2-36 from the Alfred P. Sloan Foundation to Geoffrey Hinton, Contract N00014-85-C-0133 NR 667-541 with the Personnel and Training Research Programs of the Office of Naval Research, and a grant from the System Development Foundation to Donald Norman and David Rumelhart.

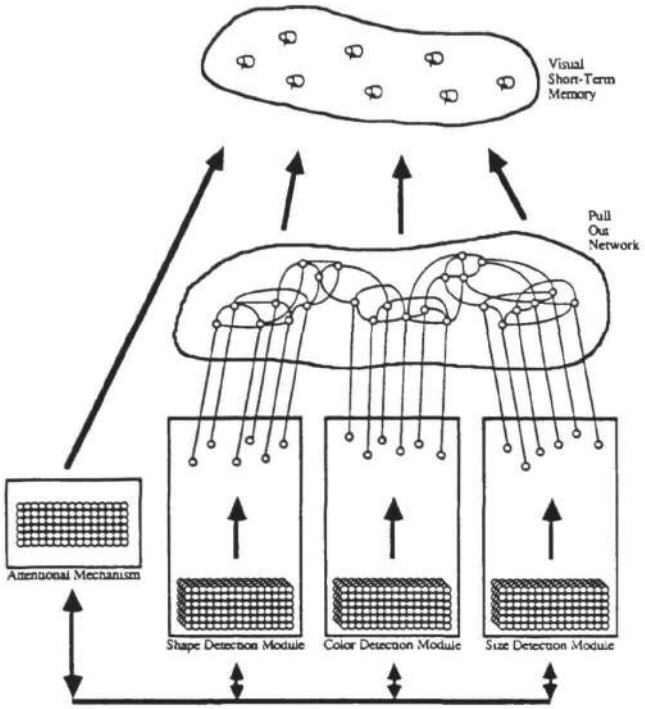


Figure 1. A sketch of MORSEL.

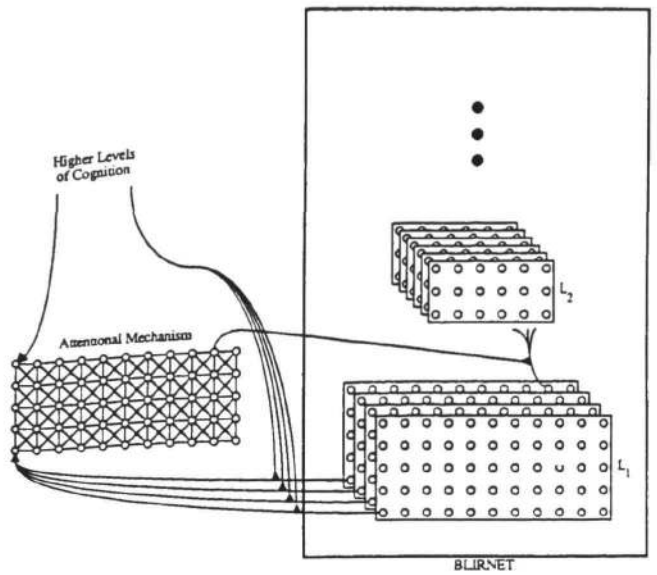


Figure 2. The attentional mechanism and its relationship to BLIRNET.

letters and bits of words, regardless of retinal location. The other processing modules of MORSEL have similar input-output properties: a retinotopic input of elementary features, and a location-independent output representation of high-level conjunctive features.

THE ATTENTIONAL MECHANISM

What might an attentional mechanism look like in the context of MORSEL? I propose a simple mechanism, one that directs a "spotlight" to a particular region of the retina (e.g., Crick, 1984; Eriksen & Hoffman, 1973; Posner, 1980; Treisman & Gelade, 1980). The attentional spotlight serves to enhance the activation of low-level retinotopic features within its bounds relative to those outside. As activity is propagated through BLIRNET and the other modules, the highlighted region maintains its enhanced status, so that in the output layer of the module, units appropriate for the attended item(s) tend to become most active as well. Consequently, these units will dominate the pull-out net competition, causing the attended item(s) to be selected. In this way, the AM allows preferential processing of attended stimuli.

The Attentional Mechanism as a Filter

The AM (Figure 2) is a set of units arranged in a retinotopic map in one-to-one correspondence with the input layer (denoted L_1) of BLIRNET. Activity in an AM unit indicates that attention is focused on the corresponding retinal location and serves to *gate the flow of activity* from L_1 to the second layer (denoted L_2) of BLIRNET. Specifically, the activity level of an L_1 unit in location (x,y) is transmitted to L_2 with probability $\xi+(1-\xi)a_{xy}$ (the *transmission probability*), where a_{xy} is the activity level of the AM unit in location (x,y) and has range [0,1], and ξ is a scaling parameter with a value of approximately .25. As long as ξ is greater than zero, the AM serves only to *bias* processing; it does not absolutely inhibit activations from unattended regions (similar to the Norman and Shallice, 1985, model).

As one might expect, highly familiar stimuli outside the focus of attention can work their way through the system better than other stimuli. To illustrate this point, BLIRNET was tested midway through training on an isolated letter recognition task. Some letters were recognized better than others: X was detected in every location and in the context of virtually any other simultaneously-presented letters, H was less consistently detected, and F even less so. Taking stability of detection to be an indication of familiarity, one might predict that performance on X should suffer less than performance on H, and H less than F, when attention is removed. This prediction is confirmed by Figure 3a. Performance here is measured as the ratio of the activation level of the target letter to the activation level of the maximally active nontarget letter, averaged over thirty presentations of the target. When this ratio falls below 1.0, the target cannot be discriminated from the nontargets. X is discriminable as long as the transmission probability is greater than .1, H .3 and F .8. Thus, BLIRNET is able to recognize familiar stimuli based on fewer perceptual features than less familiar stimuli. In other words, focal attention is less critical for highly familiar stimuli.

To further illustrate the filtering properties of the AM, BLIRNET was tested on L and G presented simultaneously. Attention was varied from being fully divided (i.e., the transmission probability was 1.0 for both letters) to being focused solely on the L (i.e., the transmission probability was 1.0 for L and 0.0 for G). Figure 3b shows that by concentrating attention on L, its relatively weak response can be improved dramatically, although this improvement is matched by a corresponding decrement in the response to G. Thus, inter-item crosstalk is reduced by focusing attention on one item. (In this example, the target:spurious activity ratio is not an absolute measure of discriminability. Because there are two stimuli, what matters for recognition are the *two* most active units. Even if a target has a ratio less than one, it may still be the second most active unit.)

System Dynamics

In the previous section, I described the manner in which a given AM state influences processing in MORSEL. In this section, I turn to the issue of how this state is computed. I begin by assuming external sources of knowledge are available that offer suggestions about where to focus. Sometimes these suggestions will conflict with one another; the task of the AM is to resolve such conflicts and construct an attentional spotlight centered on the selected location.

The AM units are interconnected to form a relaxation network that settles into states having a single, convex region of activation. The activity of each unit is updated over time as follows:

$$a_{xy}(t+1) = f\,[\,\mu \sum_{i=-1}^{1} \sum_{j=-1}^{1} a_{x+i\;y+j}(t) \;-\; \theta \sum_{\substack{i \neq x \\ j \neq y}} a_{ij}(t) \;+\; ext_{xy}(t)],$$

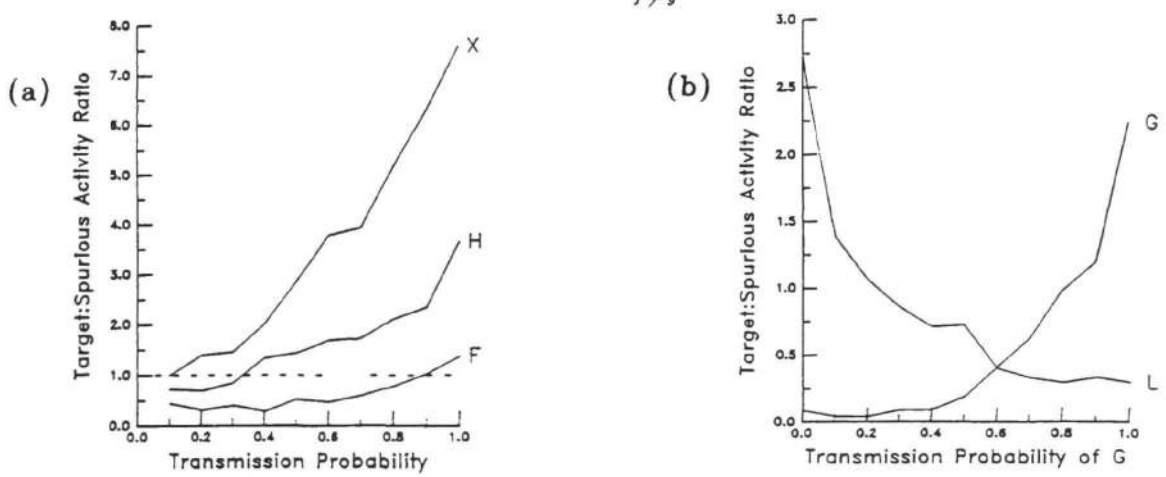


Figure 3. (a) Mean ratio of target activation to the maximum spurious (nontarget) activation for **X**, **H**, and **F** presented in location (14,2) as a function of transmission probability, averaged over thirty presentations. (b) Mean ratio of target activation to the maximum spurious activation for **L** in location (11,2) and **G** in location (20,2) as a function of attention to the **G**, averaged over thirty presentations.

where $a_{xy}(t)$ is the activity of AM unit in location (x,y) at time t, μ and θ are adjustable constants, $ext_{xy}(t)$ is an external input to location (x,y) at time t, and $f[x]$ is an identity function with saturation points at zero and one. The first term in the activation function encourages contiguous regions of activity by pushing each unit to take on the average value of itself and its eight spatial neighbors. (A neighbor is assumed to have activity level zero if it is outside the retinotopic array.) If μ is $1/9$, an exact average is computed; as a result, activation levels fade with increased distance from the center of activity. If μ is larger, however, the boundary between active and inactive regions are sharpened, so that a unit will tend to be fully on if its neighbors are on or off otherwise. The second term in the activation function limits the total activity in the network by causing each unit to inhibit all others, with θ controlling the degree of inhibition. If several discontinuous regions are simultaneously active, this term serves to suppress all but the most active region. The third term allows external sources of knowledge to drive activity in the network.

Guiding the Spotlight

These external knowledge sources can be dichotomized into two classes: *data driven* and *conceptually driven*. To consider a simple case of a "data driven" source, attention should be drawn to objects but not empty regions in the visual field. This property is incorporated into the AM by having each L_1 unit project to its corresponding AM unit (Figure 2). Similar connections to the AM should be made from the elementary feature maps of other modules, e.g., maps detecting color, texture boundaries, and motion. Through these connections, attention can be captured by such varied stimuli as an intense or flashing light, object motion, or an odd element against a homogeneous background. Further control is required, however: the mere presence of any feature should not cause an attentional shift willy nilly; attention is dependent on higher-level expectations and task demands. For example, in the task of detecting a "–" in a display of oriented line segments, one would like for only the "–" features to trigger attention. I thus propose that higher levels of cognition (*HLC*) can modulate the effect of each feature type on the AM, allowing only the features of interest to capture attention. Mechanistically, this is not difficult to implement: HLC simply need to gate the connections from each feature type in L_1 (and other such feature maps) to the AM.

Besides data-driven guidance, "conceptually-driven" guidance — direct control by HLC — is required in many situations, from reading, where text must be scanned from left to right, to a variety of experimental tasks where selection is based on location (e.g., a precue indicating the location of an upcoming target item).

If items of interest in the visual field vary in size, so must the spotlight. Empirical evidence confirms this intuition (Eriksen & Yeh, 1985; Laberge, 1983). Thus, it seems critical that HLC be able to influence not only the locus of the spotlight but also its diameter. The spotlight diameter is modulated by the parameter θ. Consequently, I assume that θ is dynamically regulated by HLC as a function of time and task.

SIMULATION RESULTS

I have implemented a simulation of the AM in which the human operator is allowed to specify the external inputs. Figure 4a presents a simple example in which two external inputs have been given, one at location (7,4) with value .2 and the other at (16,3) with value .3. Initially, activity levels of all AM units are reset to zero. Over time, the external inputs are copied into the activity of the corresponding AM units. Spotlights then begin to form around each stimulated location, but gradually activity in the region of (7,4) is suppressed, due to the fact that only one spotlight can be supported and the external input to (7,4) is smaller. By iteration 15, the network reaches equilibrium. Figure 4b shows another example with the same external inputs but θ decreased from .02 to .01. The resulting spotlight is about twice as large as in Figure 4a. One might be tempted to conclude that θ directly regulates the diameter of the spotlight, but the story is more complex, as the next example demonstrates.

In Figure 4c, the external inputs specify two blob-like regions, not individual points of activation as in the previous examples. This input pattern was constructed by presenting the stimulus **WIX MUJ** to BLIRNET (see Figure 4d), and counting the number of feature detectors active in each location of L_1. This sort of an input pattern might arise naturally on the AM if each L_1 unit fed activity into its

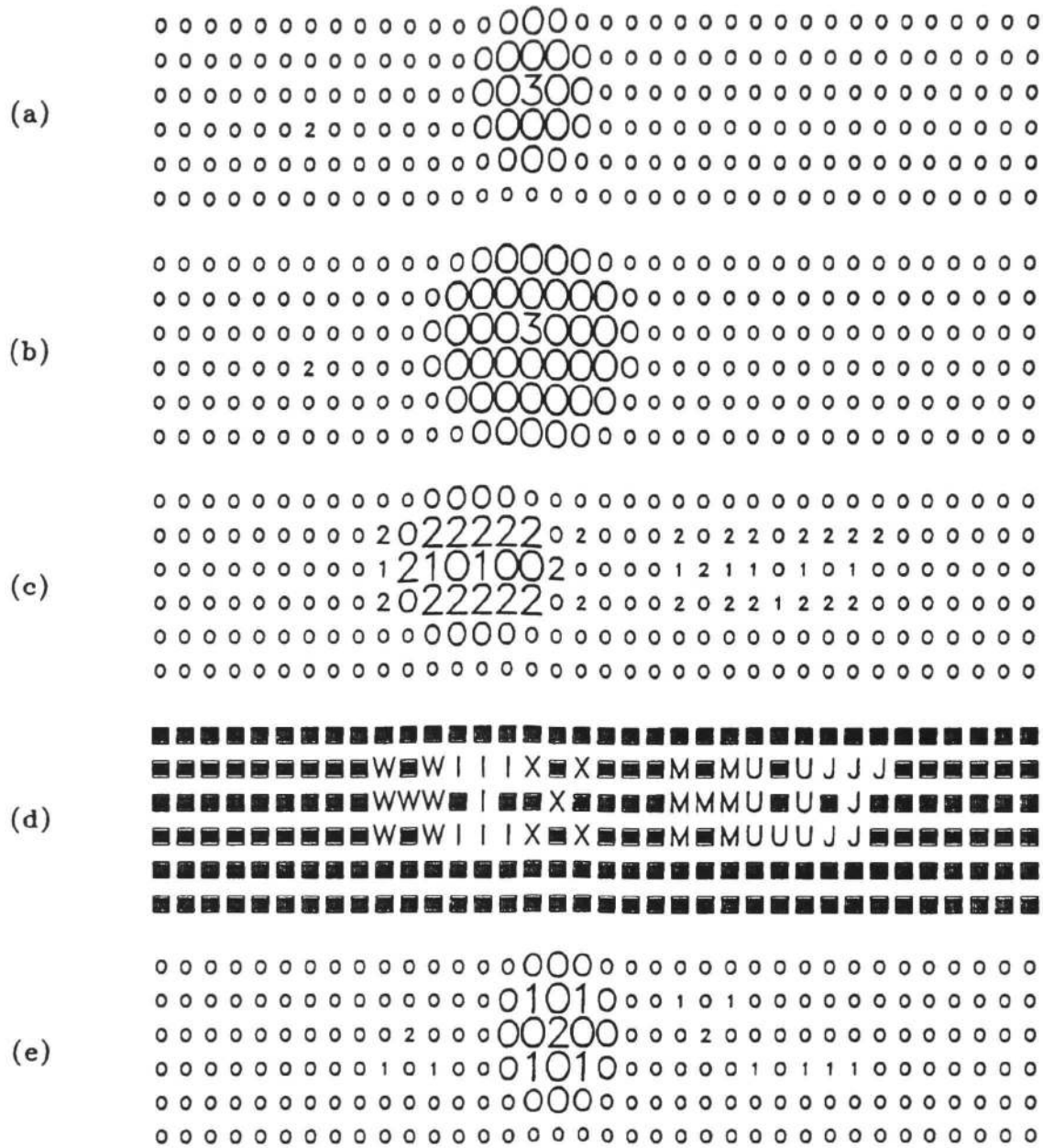


Figure 4. (a) Activity pattern in the AM at equilibrium resulting from two external inputs (μ=.22, θ=.02). The activity level of each AM unit is represented by the *size* of the digit in the corresponding position. The digit itself represents the magnitude of the external input (actually, ten times the input). (b) Activations in the AM resulting from two external inputs (μ=.22, θ=.01). (c) Activations in the AM resulting from external inputs concentrated in two regions, with slightly more input to the left region (μ=.22, θ=.02). (d) Location of the stimulus **WIX MUJ** that serves as input for Figures 4c and 4e. (e) Activations in the AM resulting from external inputs based on the \ and / features of the stimulus **WIX MUJ** (μ=.22, θ=.02). The location of the **X** is selected.

corresponding AM unit (as discussed earlier). The initial AM activity then reflects all bottom-up sources of information: attention is broadly tuned to include all items in the visual field. Over time, however, attention "narrows" on the left region — the site of WIX. This region is selected because its net external input is greater — 3.6 units of activity versus 3.5.

Although the same value of θ was used in Figures 4a and 4c, the spotlight in Figure 4c is larger. It appears that θ does not directly control the spotlight diameter. Roughly, θ can be thought of as a measure of the maximum distance allowed between two points of external activity in order for them to be enclosed within the spotlight: the larger θ is, the smaller the distance. This is a nice property of the system in that the spotlight should be on one object at a time, but it is unclear how to define the boundary of an object; with WIX MUJ, is the entire stimulus an object, is just the WIX, the X, or perhaps only one stroke of the X? θ provides one dimension along which an object's boundary can be characterized, namely, the maximum spacing between its components.

A final example of the operation of the AM is presented in Figure 4e. I have simulated the situation in which WIX MUJ is presented to BLIRNET and the L_1-AM connections are gated so that only the "\" and "/" feature maps trigger the AM. As a result, the letter X is selected. In this manner, higher levels of cognition can control which item will be selected, but only if the item has distinctive elementary features: pairs like W and M cannot be differentiated on the basis of elementary features.

THE ROLE OF ATTENTION

The AM serves MORSEL in four respects, suggesting the following roles of attention in visual information processing.

(1) *Controlling order of read out.* The AM allows MORSEL to selectively access information in the visual field by location.
(2) *Reducing crosstalk.* When items are analyzed simultaneously by MORSEL, interactions within the processing modules cause interference among items. By focusing attention on one item at a time, crosstalk can be reduced.
(3) *Recovering location information.* Remember that the output of BLIRNET — the letter-cluster representation — encodes the identity of a letter or word but not its retinal location; the operation of BLIRNET and the other modules factor out location information. However, because the current focus of attention reflects the spatial source of letter-cluster activations, the AM can convey the lost location information.
(4) *Coordinating processing performed by independent subsystems.* Each processing module operates independently of the others. Consequently, it is imperative to ensure that the results from the various modules are grouped appropriately. The AM allows this by guiding processing resources of all modules to the same spatial region. This function of attention seems analogous to that suggested by feature-integration theory (Treisman & Gelade, 1980).

EARLY VERSUS LATE SELECTION: WHERE DOES THE AM FIT IN?

A central issue in perceptual psychology over the past three decades has been the level at which attentional selection operates. Theories of attention can be dichotomized into two opposing views: *early* and *late selection*. Early-selection theories (Broadbent, 1958; Treisman, 1969) derive their name from the assertion that selection occurs early in the sequence of processing stages, prior to stimulus identification. In contrast, late-selection theories (e.g., Deutsch & Deutsch, 1963; Norman, 1968; Shiffrin & Schneider, 1977) posit that selection occurs late in processing, following stimulus identification. Additional properties go hand in hand with the central assumption of each theory (Pashler & Badgio, 1987). Early selection generally implies that (1) selection is based on low-level features such as stimulus location or color, (2) the processing system is of quite limited capacity, and (3) stimulus identification is necessarily serial. In contrast, late selection generally implies that (1) selection is based on high-level features such as stimulus identity, (2) the processing system is without capacity limitations, and (3) stimulus identification proceeds in parallel.

The view of attention presented by MORSEL is neither strictly early nor late selection. It agrees with late-selection theories in suggesting that multiple display items can be processed in parallel to a high level of representation, even to the point of making simultaneous contact with semantic knowledge (which occurs in the pull-out net). Further, selection via the pull-out net can be based on high-level — semantic or orthographic — features; this is accomplished by priming semantic units or letter-cluster units in the pull-out net to bias the pull out process. In other respects, however, MORSEL embodies an early-selection theory. First, the AM is an early selection device. It operates on a low-level representation, much in the spirit of the filtering and attenuation operations proposed by early-selection theories.

Second, the processing capacity of MORSEL is limited. If multiple items are analyzed simultaneously, interactions among the items can lead to damaging crosstalk; and there is the further problem that information about the location of each item is lost.

MORSEL thus shows characteristics of both early and late selection theories. Pashler and Badgio (1985, 1987) have proposed a similar hybrid view of attentional selection based on a large body of empirical work. Their view seems entirely compatible with MORSEL and the AM. I find it both surprising and exciting that MORSEL is in such close accord with the conclusions of Pashler and Badgio. MORSEL was not designed specifically to address attentional issues, yet it makes strong predictions concerning the nature of attentional selection. Furthermore, the hybrid view of attentional selection presented here seems like a possible resolution to the longstanding debate between proponents of early and of late selection.

In closing, I should note that Koch and Ullman (1985) have developed a related neurally-inspired model of the attentional spotlight. Their model is similar to the AM in that it consists of a topographic map in which units are activated to indicate the allocation of attention. Additionally, it operates by gating the flow of activity from a low-level input representation composed of elementary features. In Koch and Ullman's model, however, selection is performed by a simple winner-take-all network. This results in a single point of activity, as compared to the distributed activity pattern produced by the AM. Their model is thus unable to adjust the diameter of the attentional spotlight. A further drawback of the Koch and Ullman model is that it is embedded in a serial processing system, capable of processing only one item at a time. Without a system like BLIRNET, their model is merely an early selection device. This brings up the point that it is not the attentional mechanism itself that determines whether the system as a whole is best characterized in terms of early or late selection, but rather how the attentional mechanism is integrated into the rest of the system. This is where MORSEL makes a distinct contribution to theories of attention.

REFERENCES

Broadbent, D. E. (1958). *Perception and communication.* London: Pergamon.

Crick, F. (1984). The function of the thalamic reticular complex: The searchlight hypothesis. *Proceedings of the National Academy of Sciences, 81,* 4586-4590.

Deutsch, J. A. and Deutsch, D. (1963). Attention: Some theoretical considerations. *Psychological Review, 70,* 80-90.

Eriksen, C. W. and Hoffman, J. E. (1973). The extent of processing of noise elements during selective coding from visual displays. *Perception and Psychophysics, 14,* 155-160.

Eriksen, C. W. and Yeh, Y.-Y. (1985). Allocation of attention in the visual field. *Journal of Experimental Psychology: Human Perception and Performance, 11,* 583-597.

Koch, C. and Ullman, S. (1985). Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology, 4,* 219-227.

LaBerge, D. (1983). Spatial extent of attention to letters and words. *Journal of Experimental Psychology: Human Perception and Performance, 9,* 371-379.

Mozer, M. C. (1987). Early parallel processing in reading: A connectionist approach. In M. Coltheart (Ed.), *Attention and performance XII: The psychology of reading* (pp. 83-104). Hillsdale, NJ: Erlbaum.

Mozer, M. C. (1987). *The perception of multiple objects: A parallel, distributed processing approach* (Unpublished Doctoral Dissertation). University of California, San Diego.

Norman, D. A. (1968). Toward a theory of memory and attention. *Psychological Review, 75,* 522-536.

Norman, D. A. and Shallice, T. (1985). Attention to action: Willed and automatic control of behavior. In R. J. Davidson, G. E. Schwartz, & D. Shapiro (Ed.), *Consciousness and self regulation: Advances in research, Vol. IV.* New York: Plenum Press.

Pashler, H. and Badgio, P. C. (1985). Visual attention and stimulus identification. *Journal of Experimental Psychology: Human Perception and Performance, 11,* 105-121.

Pashler, H. and Badgio, P. C. (1987). Attentional issues in the identification of alphanumeric characters. In M. Coltheart (Ed.), *Attention and performance XII: The psychology of reading* (pp. 63-82). Hillsdale, NJ: Erlbaum.

Posner, M. I. (1980). Orienting of attention. *Quarterly Journal of Experimental Psychology, 32,* 3-25.

Shiffrin, R. M. and Schneider, W (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending, and a general theory. *Psychological Review, 84,* 127-190.

Treisman, A. and Gelade, G. (1980). A feature integration theory of attention. *Cognitive Psychology, 12,* 97-136.

Treisman, A. M. (1969). Strategies and models of selective attention. *Psych. Review, 76,* 282-299.

How Near Is Too Far? Talking about Visual Images

Uri Zernik and Barbara J. Vivier
Artificial Intelligence Program
General Electric Corporate Research and Development
Schenectady, NY 12301 USA

I. Visual Semantics Given as Search Directives

Lexical semantics must account eventually for the description of visual images. Ironically, existing linguistic systems are better geared to interfacing with abstract databases than to handling simple words such as **far** and **near**, **in** and **out**. Why are visual semantics elusive? Due to clarity of vision, flaws in semantic theories cannot pass unnoticed as they do in highly cerebral domains such as contract law or company takeovers. We have developed a theory, *directive semantics*, for dealing with visual descriptions.

This theory comes as an antithesis to pervasive theories of lexical semantics [4], which view language encoding/decoding as a process of constraint satisfaction. In the absence of a task domain, evaluation of theoretical results is by the linguist's ear: unusual utterances are "starred" out at convenience. In contrast, we view language processing as an objective oriented task, in which lexical semantics facilitate task performance. The testbed for the theory is a well-defined cognitive task, called "hide and seek." Linguistic utterances are used to lead a *seeing* agent toward an object *implicit*[1] in a scene.

Consider the following scenario, in which a tourist is introduced to an ancient spot in Florence.

(1) **This tall arch you see near the cathedral was designed by Pilucco two thousand years ago.**

[1] Implicit objects require computational evaluation for them to be recognized.

Sentence (1) is interpreted by the listener/viewer relative to the scene currently being perceived, the skyline of the city, in which the cathedral is prominent. Following the linguistic instruction, the viewer's eyes are guided to the cathedral, then to the arch, on which his attention is focused.

Computationally, linguistic clues such as **this**, **tall**, **arch**, and **near** are problematic for traditional linguistic systems. Consider **tall**, for example: is it a descriptive constraint? Is **tall** intended to disambiguate one particular arch by ruling out many other explicitly given candidates? Not according to the scenario above, in which no arch is explicitly identified by the viewer. **Tall** is provided as a *directive* for finding an as yet unidentified arch. **Tall** guides the viewer to search for an arch over the skyline of the city. Similarly, **near** is *not* intended as a quantifier on distance. In fact, as a quantifier, **near** is not operational in task performance—is **near** defined as 5 or perhaps as 55? Rather, **near** is provided as a search directive: start scanning the skyline from the cathedral and away! Thus, we discover the impact of lexical semantics not in abstraction but within a concrete visual task.

In this paper we describe a computational model which receives both textual and visual inputs; we explain why the direct linkage of words with geometrical description fails, and why there is a need for broader interpretation of visual relations; we show how an image understanding "agent" is guided by the linguistic text in search of perceived objects in a scene.

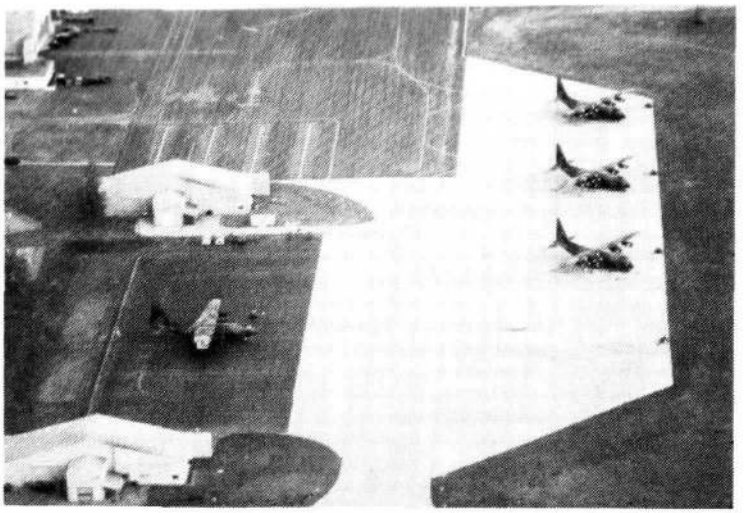


Figure 1: The Input Scene

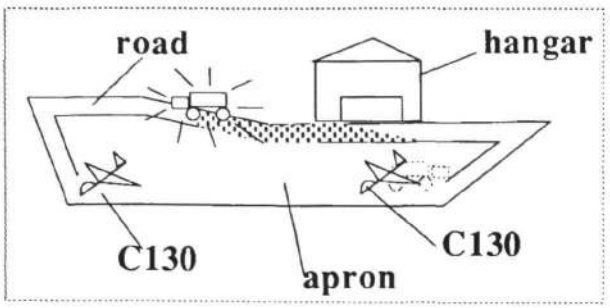
Figure 2: The Segmented Picture

A representative scene and the segmentation of the scene by the image understanding system are shown in the above figures. Figure 1 shows a photo of the Schenectady National Guard Airbase, its facilities and 3 C130 aircraft used for rescue and transportation. This picture, represented as a gray-level matrix, is converted by segmentation into a set of edges and vertices, a portion of which is shown in Figure 2. Some fixed objects in this scene are given explicitly: the hangars, the apron and the roads used by vehicles on the apron. Objects hidden (obscured by segmentation) in the picture are discovered by an image understander guided by SEER.

II. The Program SEER

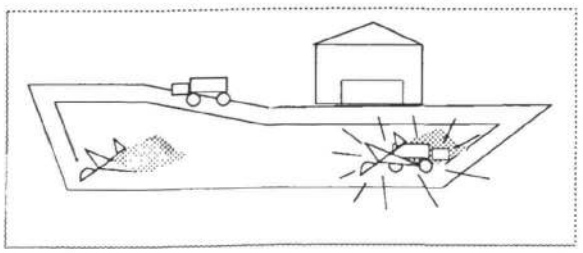
We are designing the program SEER (pronounced see'er) whose external behavior is described below.

User: **Find the fuel truck between the hangar and the C130 on the apron!**



SEER must identify a specific instance of a truck, based on references to fixed objects (i.e., the hangar, the apron). It first must find[2] the C130 aircraft, and then it searches for the truck itself based on the geometrical relation: **between the hangar and the C130**. However, **between** is not taken literally—the truck must not necessarily be on the line between the hangar and the aircraft—but it is taken as a conceptual relation. Accordingly, the search for a truck is performed along the roads connecting the two locations. As scanning of the scene progresses in search of the truck, the user receives a graphical indication of the search direction along the roads. Finally, the identified truck is illuminated.

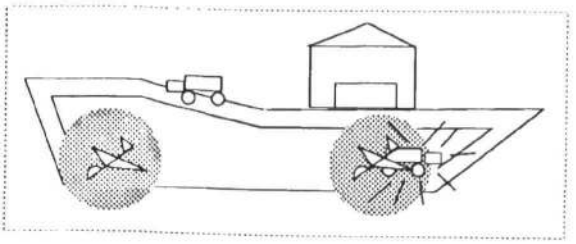
User: **Are any aircraft fueling?**



A search for a *visual script*, the C130-Fueling Script, is conducted. Accordingly, a fueling truck and a C130 in a particular spatial configuration are sought. This configuration is identified in the picture, and its constituents are marked.

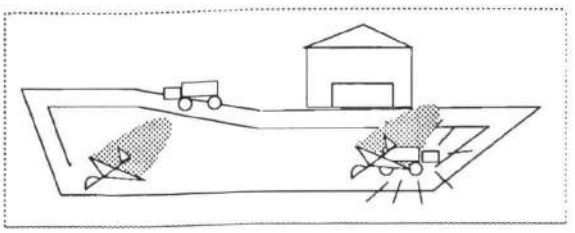
[2]In each picture, the search path is shaded; the annotations in the first drawing are for reference only.

User: **Find the truck near the aircraft.**



References such as **truck** and **aircraft** are too generic for application by an image understander. Here these references are resolved by prior *discourse*: **the aircraft** is the C130 from the first sentence, and **the truck** is resolved as either the fuel truck implicit in the fueling script or the fueling truck explicitly mentioned in the first sentence. This ambiguity is resolved by the relation **near**. The search for a truck is conducted within extending circles centered at the C130.

User: **Is the truck behind the aircraft too far?**



The relation **behind** is subjective; it must be interpreted relative to the user's *perspective*. Only by accounting for this perspective, can the program relate to the user's intended meaning. Once the truck is identified, script-based constraints are used to evaluate whether it is within the limitations of the fueling script. The last two sentences contain an apparent logical discrepancy; an instance of a truck is both "near and "too far", in the same context. This discrepancy is not real since each case has its own particular search intentions.

As demonstrated by these queries, linguistic expressions are used to direct visual scene analysis.

III. Cognitive and Technological Issues

Ultimately, cognitive models must receive and exploit input from the external world via two modalities: language and visual perception. So far, these modalities are investigated by two separate disciplines. For them to interact, a tremendous gap must be bridged between linguistic/conceptual representation of words and visual/geometrical representation of objects. In addressing this problem we choose not to ignore the main two challenges: language—*vagueness* of the medium, and vision—*brittleness* of the technology.

Consider the various interpretations of **between** in the following sequence of *locatives* [3].

(1) Point A is between points B and C.
(2) Chicago is between LA and Albany.
(3) John was watching Mary.
Barbara moved between them.
(4) My finger got stuck between the table and the wall.
(5) I find myself between a rock and a hard place.

Sentence (1), uttered by a mathematician, might possibly carry some geometrical precision. Sentence (2) uttered by an airline traveler is correct because the traveler transfers planes at Chicago en route from LA to Albany. However, it is *geometrically* incorrect: Chicago is not on the straight line constructed between LA and Albany. This discrepancy does not reflect lack of precision, which could be rectified by allowing some tolerance. It reflects the inherent complexity of linguistic relations. Sentence (3) is geometrically precise—Barbara is indeed on the line between John and Mary–but this is coincidental, due to the linear nature of light rays. The main implication of the expression is causal: Barbara "disenabled" John's seeing Mary. Sentence (4) also conveys a causal relation. Finally, sentence (5) involves two layers of metaphor: first the causal extension of **between** (X is vulnerable because he will be crushed when Y meets Z), and second, the analogy from the physical domain to another abstract domain. Thus, locatives by their nature defy direct geometrical interpretation. While allowing human speakers a great expressive power, they place on the listener the burden of identifying the appropriate implication.

Existing Image Understanding (IU) technology is quite limited. For one thing an IU program cannot identify a geometrically ill-defined object such as a person or an animal. For another, current programs cannot search for generic objects such as a cathedral or a truck. The search must be driven by a concise truck model; the slightest deviation between the designated model and the actual instance yields an identification failure. These limitations lead to three unfortunate results: (1) Without a hierarchical database (of which all models are specific instances) and gradual discrimination, all the models in the database must be exhaustively applied. All possible trucks and cathedrals are sought in each scene. (2) Without spatial relations, scenes must be scanned exhaustively. Trucks are sought in the ocean and cathedrals are sought on freeways. (3) Without conceptual expectations, there is no notion that cars drive only right-side up or that driving is oriented along the road axis. Commonsense constraints are not exploited to limit the search.

In our model, we demonstrate how these limitations of vision are overcome by linguistic inputs: (1) visual object identification is supported by linguistic reference resolution, (2) spatial relations are resolved by plan recognition, and (3) vision is driven by linguistic directives. These points are elaborated in the next sections.

IV. Resolving Locative Expressions

The resolution of locative expressions can be viewed either as a process of selection or as a process of search. We explain why, when talking about images, only the second approach is valid.

Locative Resolution as Selection: Consider, for example, the following set of expressions

[3] We define "locative" as an expression pertaining to a perceived scene. We do not use Herskovits' definition, namely, "any spatial expression involving a preposition...", since there is no method for determining whether an expression is spatial or causal.

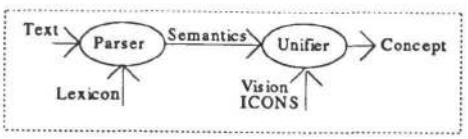
intended to identify unambiguously objects in a database:

(1) `John Marberg.`

(2) `The ship that left Long Beach to Honolulu on March 17, 1987.`

(3) `The old chair in the corner of the dining room near the picture of my parents.`

In the traditional artificial intelligence fashion such references are resolved in two steps, as shown schematically in the figure below:



(a) Convert linguistic description to semantic description; i.e., the symbol `John Marberg` is converted to a semantic template:

(person
:first-name john
:last-name marberg
:gender male)

(b) Retrieve an instance unifying templates across a database of iconic concepts; i.e., the retrieved concept in example (1) is marberg.73 whose full description in the database is:

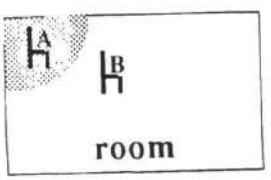
(person
:first-name john
:last-name marberg
:gender male
:age 37
:SSN 557-59-3366)

Two assumptions are made in this approach. (a) **Accessibility**: a set of concepts exists in the database in a form which lends itself to pattern matching based on features; in example (1), there is an assumed list of persons given as frame-based concepts, out of which marberg.73 is selected. (b) **Operationality**: the semantic features, given as logical predicates, are applicable in pattern matching; the strings "john" and "marberg", for example, can be validated by simple string-matching operations.

Similarly, the linguistic query of example (2) is converted to a template describing a vessel; that template is compared with all vessels in the database according to date and route; the result is a (possibly empty) set of vessels abiding by the description.

However, the two assumptions made above are not valid in certain cognitive tasks. First, as shown by Schank, Lehnert, and Kolodner[9, 1, 5], not all concepts in a semantic net are readily accessible. The main issue is not the selection of the concept but navigating in the net in order to access appropriate concepts in the first place. Second, as shown by Rosch [8], not all descriptions are crisp, and some semantic templates can be specified merely as prototypes. A prototypical dog has a tail; yet a tailless dog must still be recognized as a dog. It is yet unclear how to make prototypical descriptions operational.

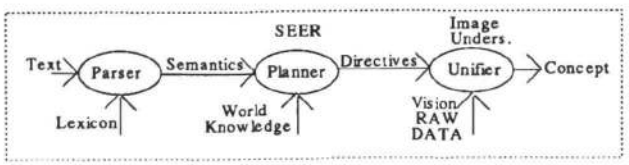
These issues are strongly manifested in the visual domain. **Accessibility**: upon receiving expression (3) above, a listener might not have accessible a list of all the chairs in the house. If really interested, the listener could walk around the house to the dining room, look at the corner and identify the chair. **Operationality**: even a simple feature such as `old` cannot become operational by a vision system. Moreover, the relation `near` is completely undefined as a quantifier. The obvious locative `in the corner` is not well-defined either as shown in the illustration below. If a corner is indeed



defined as a quarter of a cylinder, as suggested by Herskovits [4], then what is the radius of that cylinder? Is chair A "in the corner"? How about chair B?

In order to overcome these obstacles, a common practice in artificial intelligence has been to discuss "iconic vision": (a) scenes are given as collections of icons, and (b) icons are represented graphically to display the features required in pattern matching. However reality contradicts this relaxation.

Reference Identification as Search: No simplifying assumptions are made in our model about vision systems. On the contrary, we emphasize how the limitations of vision are overcome by conceptual processes. The identification task is carried out as shown below:



The scene does not contain an explicit set of icons. Mostly the scene contains raw vertices and edges. Certain objects have already been identified and precompiled. For example, the picture itself and the corners of the room have been identified in prior processing. The search is a two-step process: (a) A search planner[4] receives a sequence of search directives based on the linguistic instructions; i.e., in order to find the chair, find the dining room, the corner, the picture and then the chair. (b) The planner dispatches specific instructions to an image understander (implemented by the image understanding program [11]); i.e., `in the corner` and `near the picture` are converted into the respective directives:

(directive
  :prune OUT(corner(X))
  :order RADIAL(pos,(corner(X))))

(directive
  :order RADIAL(pos,(picture(Y))))

It must still be shown how locative expressions are converted into search directives.

[4] such as that by Hanson and Riseman[3]

V. Interpreting Locative Expressions

Three approaches to the interpretation of locatives are discussed, using the examples below:

(1) The truck is driving between the hangar and the garage.
(2) The C130 in the nosedock.

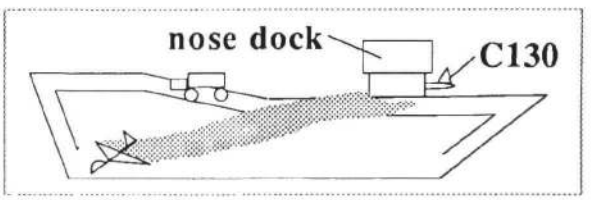
Direct Interpretation: Assuming these locatives express pure geometrical relations, a Montague-style grammar [7] can be used to convert them into geometrical primitives as in the following simplified linguistic expressions for the above examples. Thus

(1) A is between B and C.
(2) A is in B.

are converted into mathematical expressions which are based on simple geometrical primitives:

(1) `ON(A,(CONNECT(B,C)))`
(2) `CONTAIN(B,A)`

Search, following these instructions, is conducted:

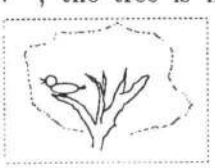


Clearly, these formulae do not capture the intended meaning: no road goes directly between B and C; the nosedock does not fully contain the C130. It is not imprecision that causes this discrepancy, but the intended meaning itself.

Geometrical Metaphor: Lakoff [6], Talmy [10], and Herskovits [4], employed geometrical metaphors in the interpretation of spatial relations. Consider, for example, the following sentence:

(9) The bird is in the tree.

As shown below [5], the tree is not really a con-



[5] the figure is taken from Herskovits' paper [4].

tainer for the bird. However, using *indirect reference* [2], the expression is reduced from 3 dimensions to 2 dimensions, meaning: the contour of the bird is in the contour of the tree, as: `CONTAIN(contour-of(tree),contour-of(bird))`

In another example, the conversion is along the whole-part dimension:

(10) John was sitting in the truck.

This example is interpreted as follows:

`CONTAIN(cabin-of(truck),john)`

This approach captures metaphors along geometrical dimensions, but it fails on metaphors which require broader world knowledge.

Object-Oriented Metaphor: The following sequence of examples shows the limitation of the geometric approach and motivates the object-oriented approach:

(10) The C130 is in the nosedock.

(11) The mouse is in the trap.

(12) John is registered in this school.

Sentence (11) can be explained by the whole-part conversion:

`CONTAIN(nosedock,part-of(C130))`

However, for a mouse's tail caught in a spring-type trap, sentence (11) is yet unexplained. The mouse is trapped, but it is not contained in the trap. Solving such relations requires knowledge about the object under analysis. The interpretation of **A is in B** depends on the nature of the interaction between A and B. Accordingly, aircraft A is **in** the nosedock B if A's nose is being maintained at dock B; mouse A is **in** trap B if A is trapped in B; person A is **in** school B if A is on the list of students of B.

The interpretation of **between** is further complicated by the *time-space duality* [6]. **The truck is between B and C** implies that a certain point in time between leaving A and arriving at B, the truck passes through it current location. This interpretation relies on the identification of a truck as a moving vehicle, whose possible location depends on its type: aircrafts are sought on air corridors, cars are anticipated on roads, and a train is probably on a railroad. Elementary world knowledge of this kind is essential to support a visual system in finding objects.

VI. Conclusions

We have examined the interpretation of locatives within a concrete task domain. In this task we have identified two central issues. First, locatives do not possess narrow geometrical interpretation, but they require world knowledge. Second, locatives are not used as simple pointers to objects—and as we have shown, their semantics is unclear when thought of as such—but they are used as directives for navigating in visual scenes. Our theory of *directive semantics* is employed in the integration of natural language and vision.

References

[1] M.G. Dyer. *In-Depth Understanding.* MIT Press, 1983.

[2] G. Fauconnier. *Mental Spaces.* MIT Press, 1988.

[3] A.R. Hanson and E.M. Riseman. Visions: a computer system for interpreting scenes. In *Computer Vision Systems*, Academic Press, 1978.

[4] A. Herskovits. Semantics and pragmatics of locative expressions. *Cognitive Science*, 1985.

[5] J. Kolodner. *Retrieval and Organizational Strategies in Conceptual Memory.* Lawrence Erlbaum, Hillsdale, NJ, 1984.

[6] G. Lakoff and D. Johnson. *Metaphors we Live By.* Univ. of Chicago, 1980.

[7] R. Montague. On the proper treatment of quantification in ordinary english. In J. Hintikka et al., editor, *Approaches to Natural Language*, 1973.

[8] E. Rosch. Principles of categorization. In B. Lloyd, editor, *Cognition and Categorization*, Lawrence Erlbaum, 1978.

[9] R.C. Schank and R. Abelson. *Scripts, Plans, Goals, and Understanding.* Lawrence Erlbaum, Halsted, NJ, 1977.

[10] L. Talmy. How language structures space. In H. Pick et al., editor, *Spatial Orientation*, Plenum Press, 1983.

[11] D. Thompson and J.L. Mundy. Three dimensional model matching from an unconstrained viewpoint. In *Proc. IEEE Robotics and Automation Conf.*, 1987.

AN ADAPTIVE MODEL FOR VIEWPOINT-INVARIANT OBJECT RECOGNITION

Peter A. Sandon
Department of Math. and Computer Sciences
Dartmouth College

Leonard M. Uhr
Computer Sciences Department
University of Wisconsin - Madison

INTRODUCTION

When we look at a familiar object from a novel viewpoint, we are usually able to recognize it. We are interested in developing a model of vision which can efficiently represent the invariant information required to recognize objects from various viewpoints, and which is capable of acquiring this information through experience. In the work described here, objects are modeled in terms of 2-D shape features. Using a hierarchical decomposition of object shapes allows parallel extraction of sub-shapes, provides storage economy for object models, and facilitates generalization of learned knowledge from one object to another.

The problem of viewpoint invariant vision can be stated as follows: The shape of an object as seen from an arbitrary viewpoint is some rigid-body transformation of visible shape features of some canonical shape of the object. Given a particular shape in an image, the vision system must recognize the object in the image as an instance of the canonical shape. To solve the recognition problem, viewpoint invariant features must be computed from the image data. Such features we refer to as being object-centered, since they have meaning relative to the object itself rather than to their image appearance. The use of object-centered features simplifies the recognition process by representing aspects of the object that do not depend on imaging parameters. In addition, object-centered representation allows more powerful generalization capabilities due to the similarity of representation within an object class.

The model is implemented in a connectionist network, in which nodes, singly or in groups, represent shape features and links represent evidential relations among shape features. We use an error correction learning method to train the network by example. The most commonly used multi-layer algorithm is the generalized delta rule (GDR). Since this algorithm is weak in a number of ways, a number of modifications to this rule have been developed and applied to this task. These modifications involve the addition of local constraints to the global error reduction constraint normally used to drive the learning. The details of these *error modification* (ErrMod) methods are reported elsewhere [Sandon 1987].

In the following sections we review some work related to our own, and then present the network model, and some simulation results. The main result involves the training of the upper layers of the network, where the desired generalization across translations of objects is demonstrated. This generalization capability eliminates the need to expose the system to every object under every transformation in order to obtain complete recognition.

RELATED WORK

Two related models that have been described in recent years [Hinton 1981, Ballard 1984] attempt to cooperatively identify the transformation and recognize the canonical shape simultaneously. A partial identification of the transformation can be used to constrain the possible interpretations of the object shape and vice versa. In the present work, we combine the representation of transformation information suggested in the Hinton and Ballard models with the hierarchical representation of shape used by Uhr [1972]. A major constraint on the resulting model is that it be amenable to learning of both shape and transformation.

Uhr's Recognition Cone

The recognition cone is an example of a parallel-hierarchical vision model. The recognition cone consists of layers of *transforms*. Each transform produces an output value that is some characteristic function of its input values. In general, the output values of each transform are used as input values to other transforms. The transforms in the lowest layer compute properties of the image pixels, those in the next lowest layer compute properties of these lowest layer transform properties, and so forth. Higher layers in the recognition cone are logarithmically smaller than lower layers, giving a hierarchical, pyramidal structure. Information is converged as it proceeds up the cone, so that higher level transforms, performing local operations on the information below, compute more successively global properties of the image than do transforms at lower layers. The properties computed by any given transform are stored locally, and/or passed up to be accessible to transforms at higher layers. In addition to the bottom-up processing, the property computed by a transform can be passed down the pyramid to provide feedback to earlier layers.

The recognition cone has much in common with the connectionist paradigm. It can be described as a connectionist network by making the following correspondence. Each transform is implemented by a single node in the network. A given transform accesses the result of another transform through a connection. The connectionist interpretation is more restricted than the original recognition cone in the complexity of transforms that can be directly implemented, in the use of local memory and in the control structure that is used. The local and layered nature of the recognition cone transforms, however, makes them amenable to this connectionist interpretation.

The Hinton / Ballard model

An approach which *explicitly* represents both the transformation invariant shape and the transformation itself is that proposed by Hinton [1981] and extended by Ballard [1984]. In this connectionist model, processing units are grouped into three distinct sets (see Figure 1), referred to as the retina-based frame, the object-based frame and the mapping units. The retina-based units represent features extracted from the image with spatial relations represented relative to the imaging device. The object-based units represent spatial relations relative to the object, without regard to the particular image representation. The mapping units represent the

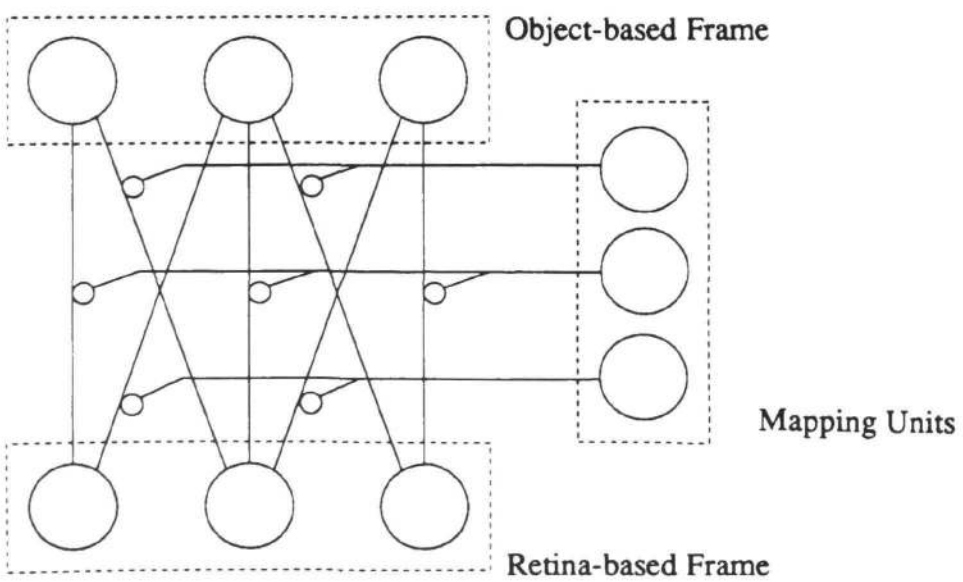


Figure 1

transformation between the retinal shape and the canonical coordinate frame represented in the object-based frame.

By explicitly representing the one transformation associated with an object, the network implements what Hinton refers to as the single viewpoint constraint. Retina-based units compute features based on properties of the image itself. Object-based units combine retina-based and mapping unit features to compute object-centered features. The small circles in Figure 1 represent the conjunctive modification of the retina-based to object-based connection by the mapping units. Similarly, mapping units use information from both retina-based and object-based units to compute the current object transformation. This computation is not represented in the figure. The interdependence of the object-based and mapping units requires a cooperative computation in which a partial result in one set of units improves the result in the other set.

NETWORK MODEL

The basic structure of our model is that of a pyramid-like primary network in which shape features are hierarchically represented, augmented with a secondary 'context' network in which transformation information is represented. The transformation information is conjunctively combined with shape features, at various levels of the pyramid, to produce representations of shape that are successively more object-centered in higher layers of the network. The gradual transition from strictly retina-based to strictly object-based features has two advantages. First, the increased connectivity due to conjunctive combination is spread over several layers. Second, local connectivity is maintained, which allows the recognition cone to discriminate features of locally, and successively more globally, interacting objects. We refer to all layers of the shape pyramid below the point where context information is introduced as retina-based layers. All layers above any use of context information are object-centered layers. Those layers in between are transition layers.

The particular instantiation of this model that has been simulated is a 2-D translation network which succeeds in recognizing various stick-figure patterns under all translations within a small image plane. (see Figure 2). Layer 0 is the input image. Shape features are represented in layers 1 through 4, with layer 5 representing objects to be recognized. The transformation (in this case, translation) of the object is represented in layers 1^c through 4^c. The outputs of the upper two layers of the context network are conjunctively combined with shape features at layers 3 and 4 in the shape hierarchy.

In the Hinton model, it is suggested that both shape and transformation can be extracted from the image. This requires that the identity of the shape be used to define the transformation, and that the identity of the transformation be used to define the shape. These mutually dependent definitions of shape and transformation require feedback paths and a relaxation process for computation by a network. In order to maintain the feedforward structure of the network, we do not include the feedback term from the object-centered features in computing the transformation. For the objects used in our simulations, the network is able to compute the transformation without this feedback term.

Each layer in the network consists of a compact square array of competing node clusters. A cluster is composed of a number of nodes, typically from 2 to 9, which compete for adjustments to their connection weights through the error modification learning mechanism. Each node is implemented as a logistic processing element [Rumelhart, et. al. 1986]. The input image is 15 pixels on a side. Layers 1-5 and 1^c to 4^c are composed of 13, 11, 5, 3, 1, 9, 7, 3 and 3 clusters on a side, respectively. The size of the cluster in the output layer (layer 5) ranges from 6 to 50 in various simulations.

Due to the complexity of this network and the various characteristics of the network structure and learning algorithms to be demonstrated, we have simulated separate pieces of the network, which we now describe.

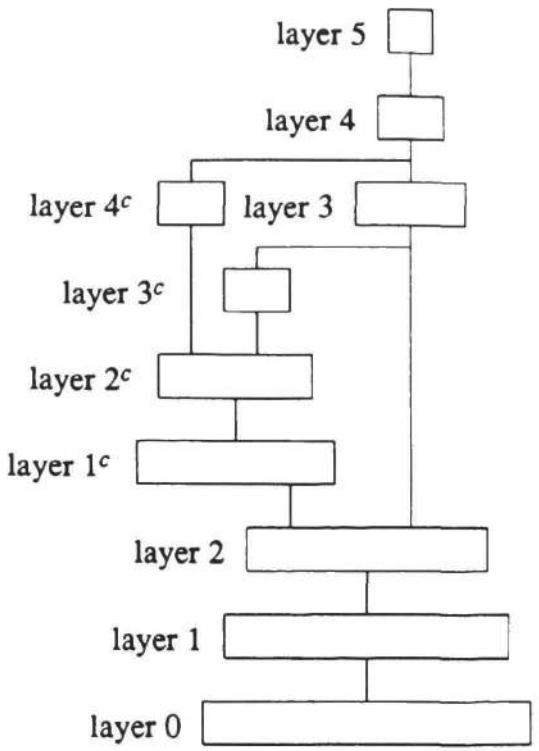


Figure 2

EXPERIMENTS

Learning object-centered representations

The first learning simulation uses the top three layers of the shape pyramid (layers 3-5), and the top layer of the context network (layer 4^c). This simulation demonstrates that object recognition can be learned through the development of object-centered feature detectors and explicit representations of location information.

To simulate this 3-layer network, we provide input directly to layers 3 and 2^c. Patterns are provided to various intermediate layers of the network in the experiments described. The patterns chosen are consistent with a particular interpretation that can be represented by the nodes at that layer, though many other representations are possible. One of six 'letter' patterns (see Figure 3) in one of nine locations is presented in layer 3. A single activation of one of nine nodes in layer 2^c represents the location information for the context subnetwork. Using these six shape patterns and nine locations yields a set of 54 images to be presented to the network. These are split into a set of 27 training patterns and 27 test patterns. A response is considered correct if the maximally active output node corresponds to the correct pattern class and has an output value greater than .5.

In the first experiment, learning of the 27 training patterns proceeds quickly, yielding 89% performance in 500 cycles using GDR. However, when the remaining 27 patterns are presented to the network as a test set, only 4 are correctly classified. This combination of relatively fast learning and weak generalization indicates that the capacity of the network to store patterns is high compared to the number of patterns to be stored. This allows the network to perform well by representing each pattern individually, rather than as a set of shared features based on the regularities intrinsic to the pattern collection. The result is rote learning, which lacks generalization capability. To overcome this problem, the capacity of the network is limited in the succeeding experiments by using only six of the eighteen nodes of layer 4.

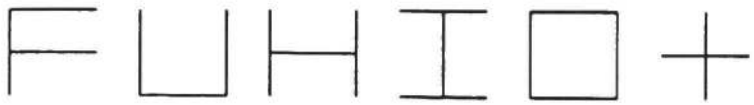


Figure 3

Repeating the training experiment using GDR on the 27 training patterns results in 75% recognition performance after 5000 cycles. Error modification improves this result to 96% performance after 5000 cycles. The reduced capacity of the network has increased the difficulty of the learning task, as expected. However, in the simulation using error modification, only two of the nine context nodes are utilized, in the sense that they are active for some patterns and not for others. Since the location information is implicit in the image, the primary network is capable of solving much of this task without the explicit representation of location supplied by the context network. This will not lead to the desired object-centered features, however, which are generalizations over object locations. The results of applying the adapted network to the test patterns are again 4 of 27 correct.

In the next experiment we use the full set of 50 shape patterns and 9 locations for a total of 450 image patterns. In order to achieve good recognition performance on this task, the network must represent pattern features more efficiently than in the previous experiment. We use a training set of 150 patterns leaving 300 patterns as a test set. The set of 150 training patterns includes 3 presentations of each of the 50 fixed patterns in three different locations.

The increased number of patterns makes this task more difficult than the previous one. Using GDR, only 12% of the patterns are correctly recognized after 20000 cycles of training. Using error modification, performance on the training set reaches 100% after 12000 cycles (80 presentations per pattern). Table 1 presents the results of running a set of 10 testcases using GDR and error modification. In the GDR simulation, the network did not learn to correctly classify enough of the training patterns to demonstrate any significant generalization. In the error modification simulation, seven of the ten testcases resulted in perfect classification of the training patterns, and each of those testcases demonstrated a strong generalization to the test patterns.

Table 1 - Learning performance on training and test patterns
% performance after 20000 cycles

Testcase	GDR		ErrMod	
	training	test	training	test
A	37	6	100	100
B	40	6	90	44
C	35	6	76	23
D	38	9	100	99
E	31	11	100	100
F	38	8	100	100
G	42	6	92	40
H	31	10	100	100
I	30	8	100	91
J	39	11	100	100

Generalization allows unknown patterns to be correctly classified, as demonstrated above. In addition, generalization leads to efficient learning since some knowledge of learned patterns is transferred to related unknown patterns. There are a number of ways to demonstrate this transfer. Table 2 presents the number of cycles needed to reach maximum performance for various training set sizes, using testcase A. If no generalization takes place, we expect the time needed to train the network to increase as the size of the training set increases. However, Table 2 shows that the time needed to train the network on all 450 patterns is less than that needed to train on 100 patterns. This indicates strong generalization among patterns which enhances the learning of additional patterns which fit the generalization.

The set of experiments described in this section demonstrates the key network capability that is desired for recognizing familiar objects from novel viewpoints.

Other sub-networks

We now describe additional simulations that demonstrate the behaviors of other sub-networks.

Context. To demonstrate the learning of context information from shape features we simulated the network composed of layers 0, 1, 2, 1^c and 2^c of the 2-D Translation network. Patterns are presented as activations of the input nodes of layer 0. The weights of the nodes in layers 1 and 2 are predetermined to extract simple line features from the image. The input patterns presented are those corresponding to 17 of the 50 shapes and 25 of the 49 locations used in the previous simulation. The desired output, at layer 2^c is to have a single active node corresponding to one of 49 locations of the shape in the image.

This two-layer learning task turns out to be fairly easy for the GDR algorithm. The network is able to reach 100% performance in 4500 cycles.

Retinotopic. The second simulation involves the lower layers of the network, where feature representations are purely retina-based. These layers are 4 and 5 layers removed from direct training. This results in very weak training signals from the layers above. For this reason, we apply the methods of error augmentation [Sandon 1987], which combines top-down error driven learning with bottom-up stimulus driven learning.

For this simulation, we adapt the first four layers of the shape network. Input patterns are presented at layer 0. At layers 1 and 2, an error augmentation algorithm is used to adapt the weights. At layers 3 and 4, GDR is used. Training input is provided directly to layer 4.

Using a set of 100 patterns, each consisting of one of 47 shapes at one of 25 locations, the performance of this network reaches 61% after 20000 cycles. The significance of this simulation is in demonstrating that the low level features, developed mostly through a bottom-up process, are sufficient to produce the features at higher layers that are required by the translation-invariant recognition task. Using the self-organizing component alone in the lower two layers yields a performance of only 11%.

Transition. The final simulation concerns the gradual transition of retina-based to object-based features. For this purpose, we simulate layers 2, 3, 4, 2^c, 3^c and 4^c of the 2-D Translation network. Inputs are supplied

Table 2 - Learning performance of ErrMod
Maximum performance
and cycles needed to achieve it

size	% performance	cycles(x1000)
50	99	20
100	96	20
150	100	12
200	100	14
450	100	10

directly to layers 2 and 2^c, while training input is provided to layer 4.

Learning in this sub-network is difficult because it involves two context layers comprising one two-layer and one three-layer backpropagation path. Most previous work using error-correction learning in layered networks has been applied to two-layer networks without conjunctive connections. Error modification is used at both context layers to obtain sufficient differentiation of function among the nodes to allow learning to take place. In addition, error augmentation is used in the context layers due to the length of the backpropagation path.

This simulation uses 100 input patterns each consisting of one of 47 shapes and 9 locations. After 20000 cycles, network performance reaches 80%, but it does not improve during an additional 10000 cycles. Although the combination of learning algorithms leads to only 80% performance on this task, the result is encouraging considering the difficulty of the task.

CONCLUDING REMARKS

We have described a network model of shape classification that is capable of learning to recognize objects under various translations, including novel ones. The 2-D translation network successfully learns to "recognize familiar objects from novel viewpoints" by developing object-centered representations of the shapes through the explicit representation of location. These transformation-invariant features support the necessary generalization of shape information across locations.

In the simulations that have been described, we have had to apply input and training patterns at intermediate layers of the network. Our ability to define such patterns may imply an understanding of this particular problem which obviates the need for learning. However, we hypothesize, and intend to demonstrate, that the same network structure can be applied to a problem involving non-rigid transformations, such as recognition of handwritten letters. In such a task, the choice of *a priori* representations for each layer of the network would be extremely difficult, making learning a crucial part of the modeling process.

REFERENCES

- Ballard, D. H., "Parameter Nets," *Artificial Intelligence* **22** pp. 235-267 (1984).
- Hinton, G. E., "A Parallel Computation that Assigns Canonical Object-Based Frames of Reference," *Proc. 7th IJCAI*, pp. 683-685 (1981).
- Rumelhart, D. E., G. E. Hinton, and R. J. Williams, "Learning Internal Representations by Error Propagation," pp. 318-362 in *Parallel Distributed Processing Volume 1*, ed. D. E. Rumelhart and J. L. McClelland (eds.),Bradford Books, Cambridge, MA. (1986).
- Sandon, P. A., "Learning Object-Centered Representations," PhD. Dissertation, Univ. Wisconsin - Madison (August 1987).
- Uhr, L., "Layered 'recognition cone' networks that preprocess, classify and describe," *IEEE Trans. Comput.* **21** pp. 758-768 (1972).

Spatial Reasoning Using Sinusoidal Oscillations

Ian Pratt
Department of Computer Science
Victoria University of Manchester
Manchester, M13 9PL
England

Abstract

This paper outlines some preliminary results concerning the use of sinusoidal oscillations to represent vectors in two-dimensional space. The proposed representation scheme permits efficient implementation of translation and rotation, and immediate detection of such relations as collinearity and proximity of points. This scheme is then extended so that arbitrary convex regions of the plane are represented using a pair of signals varying over time. Finally, the advantages of representing convex regions in this way are shown to derive from the resulting ease with which such regions can be translated and rotated in the plane, and—more strikingly—from the simplicity of determining whether two such regions overlap.

1 Introduction

This paper presents an approach to spatial representation such that that: (i) translations and rotations of points and regions can be readily performed; (ii) relations such as collinearity and proximity of points can be efficiently detected; and (iii) the question of whether regions overlap, and, if so, by how much, can be settled with a minimum of computation. For reasons which will become evident, we concentrate on the problem of representing *convex* regions of the *plane*.

The proposed representation system is based on a well-known correspondence between sinusoidal oscillations with a fixed frequency ω, and vectors in two-dimensional space. Section 2 describes how we can represent a vector (equivalently, a point in space), by having a device which emits an appropriate sinusoidally varying signal. (Here, "device" may be either a physical device or a virtual device—i.e. a programming construct.) Section 3 extends these results to show how we can represent any convex region of the plane using a device emitting a pair of signals varying with some fixed time period. Such regions, thus represented, can be efficiently translated and rotated; and, as we see in section 4, given two such regions, it is a straightforward matter to determine whether they overlap. As such, I claim, the proposed

system promises to combine the expressive power of numerical co-ordinate systems and their variants, e.g. [2], [3], with the computational tractability of qualitative, object-based spatial representation schemes such as [1].

2 Representing vectors

There is a natural correspondence between the set of sinusoidal oscillations with some fixed frequency ω and the set of vectors in two-dimensional space. Given a sine wave

$$w(t) = a\sin(\omega t + \phi)$$

with frequency ω, amplitude a and phase-lead ϕ, we can associate the vector v whose polar co-ordinates are (a, ϕ).

The key feature of this correspondence is that, given two vectors, the *superposition* of their corresponding sinusoidal waves yields a third sinusoidal wave, also of frequency ω, which corresponds to the vector sum of the original two vectors. More formally:

Theorem 1: Let $w_1(t) = a_1\sin(\omega t + \phi_1)$, $w_2(t) = a_2\sin(\omega t + \phi_2)$ be sinusoidal waves of frequency ω. Let v_1, v_2 be the corresponding vectors (i.e. having polar co-ordinates (a_1, ϕ_1), (a_2, ϕ_2) respectively). If $w_3(t)$ is the pointwise sum $w_1(t) + w_2(t)$, and if v_3 is the vector sum $v_1 + v_2$, then $w_3(t)$ is also a sinusoidal wave of frequency ω, $w_3(t) = a_3\sin(\omega t + \phi_3)$, where (a_3, ϕ_3) are the polar coordinates of the vector v_3.

Textbooks on the theory of AC circuits use this result to calculate the effects of combining sinusoidally varying currents and voltages by adding the corresponding vectors. Here, however, rather than using vectors to reason about sine waves, I propose that we use sine waves to reason about vectors.

If we have a device (physical or virtual) continuously emitting a signal varying sinusoidally over time with frequency ω, that device can represent a vector. The amplitude, a, of the wave represents the length of the vector, and the phase lead, ϕ (measured relative to some internal clock), represents its orientation. The varying signal may be a sequence of numbers, or a continuously varying potential difference, or yet other things besides. In this paper, I leave the details of the implementation open, assuming only that suitable sine waves can be readily produced, superposed, and compared in simple ways.

Now suppose we have two such devices running in parallel and emitting sine waves of frequency ω, representing, respectively, vectors v_1 and v_2. By piping those signals to a unit which superposes (i.e., pointwise adds) them—call it an *addition unit*—we have, in effect, computed the vector sum $v_1 + v_2$ (fig.1).

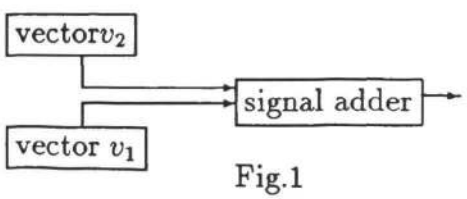


Fig.1

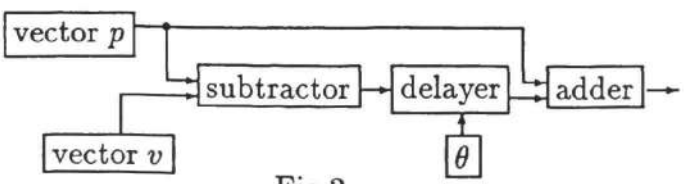


Fig.2

Similarly, *mutatis mutandis*, for vector subtraction: by piping the two sinusoidal waves to a ***subtraction unit*** which pointwise subtracts one from the other, we have computed $v_1 - v_2$.

To rotate a vector v, thus represented, clockwise through some angle θ (about the origin), we need only increase, by θ, the phase-lead of the wave used to represent v. This corresponds to *delaying* v's output signal by the amount of time $(2\pi - \theta)\omega^{-1}$ (ignoring the first oscillation), again, an operation which I shall take it can be simply performed. To rotate a vector about a point p (other than the origin), it suffices to find $v - p$ (obtainable by piping v and p into a vector subtraction unit), rotate the result through θ about the origin (as just described) and add p (by piping the result together with p to a vector addition unit). The process is illustrated in fig.2.

It is also easy to test for such relations as sameness of orientation, collinearity and proximity. Two vectors have the same orientation if and only if their corresponding sine waves are in phase; so a device able to check that the maxima of two waves coincide could perform this test. To test for the collinearity of v_1, v_2 and v_3, it suffices to compute both $v_1 - v_3$ and $v_2 - v_3$ and then to determine whether these two vectors have the same or opposite orientation. Thus, the arrangement depicted in fig.3 would suffice for such an operation.

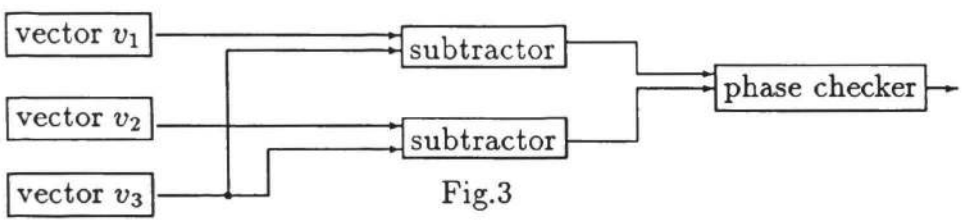


Fig.3

Similarly, two points, corresponding to vectors v_1 and v_2, will be *near* each other if and insofar as the length of $v_1 - v_2$ is small. Therefore, detecting the proximity of points just involves piping the corresponding signals to a subtraction unit and then piping the result to a device which registers the maximum amplitude of an incoming signal.

3 Representing convex regions

Let us begin with a simple example. Consider the line AB between two points (vectors) v_A, v_B, as shown in fig.4 (the sine-wave representation of v_A, v_B is also given).

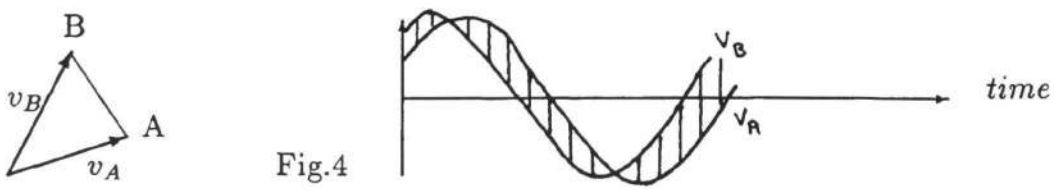


Fig.4

The set of vectors lying on the line AB is the set $\{\kappa v_A + \lambda v_B | \kappa, \lambda \in [0,1], \kappa + \lambda = 1\}$. Now there is a correspondence between this set and the shaded area of fig.4: let $w(t)$ be any sine wave (of frequency ω) and let v be its corresponding vector; then $w(t)$ lies within the shaded region of figure 9 if and only if v lies on the line AB. The task of this section is to generalise this result to show how more complicated regions can be represented.

Definition: A set S of vectors is said to be *convex* iff, $\forall x, y \in S$, $\forall \kappa, \lambda \in [0,1]$ s.t. $\kappa + \lambda = 1$, we have $\kappa x + \lambda y \in S$.

Fig.5 shows some more convex sets of vectors (the shaded regions, *not* just their boundaries) accompanied by the sinusoidal representations of some of the key points on their peripheries. (Points in the plane are identified with the corresponding vectors in the obvious way.)

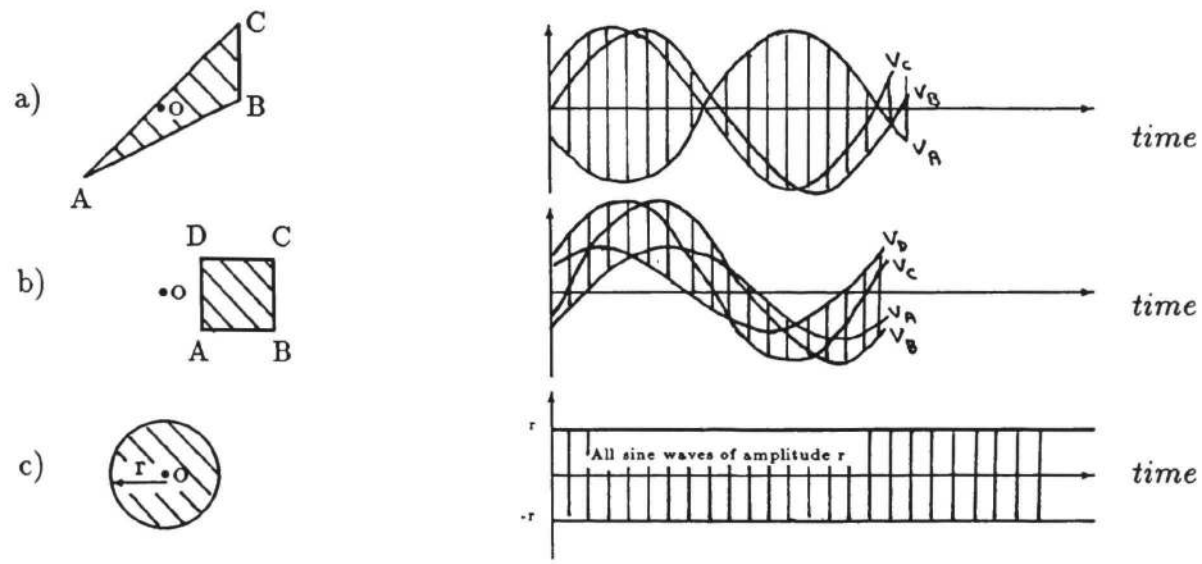


Fig.5 ("o" indicates the origin)

We shall presently show that the shaded regions in the two sets of diagrams correspond, in a sense we now proceed to define.

Definition: If V is a set of vectors, define the *convex hull* of V, to be the smallest convex set containing V.

Definition: If W is a set of sine waves $w(t)$ of frequency ω, define the *sinusoidal hull* of W, to be the pair of functions: $\Sigma(t) = \langle a(t), b(t) \rangle$, where $a(t)$, $b(t)$ are functions defined by: $a(t) = \sup_{w \in W} w(t)$ $\quad b(t) = \inf_{w \in W} w(t)$.

$a(t)$, $b(t)$ represent the upper and lower bounds, respectively, of all the sine waves in W; if these upper or lower bounds do not exist, we set $a(t) = \infty$, $b(t) = -\infty$, as appropriate.

Definition: If $\Sigma(t) = \langle a(t), b(t) \rangle$, is a sinusoidal hull, and $w(t)$ is a sine wave, we write $w \preceq \Sigma$ (read *w is contained in* Σ) iff $\forall t$, $a(t) \geq w(t) \geq b(t)$.

We can now state the main theorem of this section:

Theorem 2: Let V be a closed set of vectors, and let W be the set of sine waves corresponding to the vectors in V. Let S be the convex hull of V. Let Σ be the sinusoidal hull of W. If v is any vector, and $w(t)$ its corresponding sine wave, then $v \in S$ iff $w \preceq \Sigma$.

In other words, the sinusoidal hull of a set (finite or infinite) of sine waves corresponds to the convex hull of the set of vectors those sine waves represent.

Hence, in order to represent the convex hull of a finite set of vectors, all you need to do is to take the sine waves corresponding to those vectors and pipe them to two devices, one outputting the (pointwise) maximum of all its incoming signals, the other the minimum. (Thus these signals correspond to the functions $\langle a(t),\ b(t)\rangle$ respectively of some sinusoidal hull Σ.) Fig.5a shows roughly what those outputs would have to be in order to represent a triangle. Fig 5b does the same for a square. Such a pair of signals can be used to determine whether any given vector v, represented by sine wave $w(t)$, is in the region in question simply by checking whether $w \preceq \Sigma$ (i.e, whether $a(t) \geq w(t) \geq b(t)$ holds over an entire oscillation).

Not that a sinusoidal hull *need* be generated from a finite number of sine waves in this way: fig.5c shows how the sinusoidal hull corresponding to a circular region centred at the origin is easily generated — it is just a pair of constant signals. But it cannot be generated from any finite number of sine waves. Similarly, for other convex, curvilinear shapes.

Furthermore, *any* device emitting a pair of periodic signals $a(t)$ and $b(t)$, with period $2\pi\omega^{-1}$, can be taken as representing a (possibly empty) convex set. To see this, note that there will either be a maximal sinusoidal hull that lies between these two signals, or else no such sinusoidal hull at all, where a *maximal* sinusoidal hull is one for which the set of sine waves it contains is maximal. And since that maximal sinusoidal hull (if there is one) will pick out some convex set or other, then the original signals, $a(t)$ and $b(t)$, can themselves represent that convex set. As before, to test whether a vector is in that set, we take its corresponding sine wave $w(t)$ and check whether $a(t) \geq w(t) \geq b(t)$ holds over an entire oscillation.

Suppose we want to translate a convex set C by a vector v, where C is represented by the signals $\langle a(t),\ b(t)\rangle$ and v by the signal $w(t)$. Then the translated convex set will be represented by $\langle a(t) + w(t),\ b(t) + w(t)\rangle$. Hence, the translation of convex regions, like that of single points, can easily be effected by means of piping the appropriate signals to an addition unit as described above. Similarly, the rotation clockwise through an angle θ of a convex region about the origin can be effected by delaying the signals $a(t)$ and $b(t)$ by an amount of time $(2\pi - \theta)\omega^{-1}$.

4 Detection of Overlaps of Convex Regions

To detect overlaps, the following definitions will prove expedient:

Definition: Suppose S_1, S_2 are sets of vectors. Define their *inner difference*, $S_1 \ominus S_2$ to be the set $\{(x - y) | x \in S_1, y \in S_2\}$.

Definition: Suppose $\Sigma_1(t) = \langle a_1(t), b_1(t)\rangle$ and $\Sigma_2(t) = \langle a_2(t), b_2(t)\rangle$ are sinusoidal hulls. Define their *inner difference*, $\Sigma_1 \ominus \Sigma_2$ to be the pair of functions $\langle a_1(t) - b_2(t), b_1(t) - a_2(t)\rangle$.

If S_1, S_2 are convex, so is $S_1 \ominus S_2$. Also, $S_1 \cap S_2 \neq \emptyset$ iff $0 \in S_1 \ominus S_2$. Hence, to determine whether S_1 and S_2 overlap, it suffices to determine whether $0 \in S_1 \ominus S_2$. If Σ_1, Σ_2 are

sinusoidal hulls, so is $\Sigma_1 - \Sigma_2$. It is formed by subtracting the lower bound of the second hull from the upper bound of the first, and the upper bound of the second hull from the lower bound of the first. Inner differences of sinusoidals hulls can therefore be computed using subtraction units of the kind encountered above. Also, inner differences of sinusoidal hulls correspond to inner differences of convex sets:

Theorem 3: If S_1 and S_2 are convex regions corresponding to sinusoidal hulls Σ_1, Σ_2 respectively, then $S_1 \ominus S_2$ is a convex region corresponding to the sinusoidal hull $\Sigma_1 \ominus \Sigma_2$.

Whence:

Theorem 4: Let w_0 denote the zero-amplitude sine-wave: $w_0(t) = 0$ for all t. If S_1 and S_2 are convex regions corresponding to sinusoidal hulls Σ_1, Σ_2 respectively, then $S_1 \cap S_2 \neq \emptyset$ iff $w_0 \preceq \Sigma_1 \ominus \Sigma_2$.

Since, then, we have a method for computing the inner difference of two sinusoidal hulls, and a means of determining whether a sinusoidal hull contains the zero wave, we can immediately detect whether two convex regions overlap by piping their outputs to an appropriate arrangement of devices.

A caveat. Theorem 4 should be applied with care. Recall from section 3, that *any* pair of periodic signals (with period $2\pi\omega^{-1}$) can represent a (possibly empty) convex set. Given two such pairs of signals, $\langle a_1(t), b_1(t)\rangle$ and $\langle a_2(t), b_2(t)\rangle$, representing S_1, and S_2 respectively, we can define their inner difference in the same way as for sinusoidal hulls. But it is in general *false* that the resulting pair of signals $\langle a_1(t) - b_2(t), b_1(t) - a_2(t)\rangle$ will contain w_0 only if $S_1 \cap S_2 \neq \emptyset$. Theorem 4 requires that $\langle a_1(t), b_1(t)\rangle$ and $\langle a_2(t), b_2(t)\rangle$, be sinusoidal hulls, as defined in section 3.

5 Discussion

On the present view, a complex two-dimensional scene is represented by a collection of "devices", each broadcasting either a single sine wave (representing a single point) or a pair of periodically varying signals (representing a convex region). By piping the outputs of these devices through suitable transforming units, geometrical operations like translation and rotation can be performed. Moreover, by piping the same outputs through yet other devices, tests for relations like collinearity and proximity of points and overlap of regions can be made. The computation involved in these operations and tests is trivial, in marked contrast to the nightmarish problem of determining intercepts of curves expressed as equations in, say, a Cartesian co-ordinate system. It is as if the sinusoidal representation "flattens out" the problem of determining the membership of a point in a convex region, yielding the easier problem of determining whether one signal lies between two other signals over a period of time.

It will be objected that this assessment ignores the difficult task of converting back and forth between a Cartesian co-ordinate representation and the sinusoidal hull representation.

But this would be to miss the point of the present proposal. That proposal is not that regions be represented by equations of bounding curves, and that those equations be converted into sinusoidal hull signals in order to perform transformations and tests on them. Rather, it is that the equations of bounding curves can be dispensed with altogether: all we have is the collection of signal-emitting devices. Thus, to know what a square is, or a triangle or a circle or an oval, our system would have to be able to activate a device outputting the appropriate signals, not state an equation in Cartesian co-ordinates (of course: the device in question might have to store its signals as a pair of equations $a(t) = ...$, $b(t) = ...$). To repeat: the policy is not to augment traditional co-ordinate representations, but to dispense with them.

So conceived, the fate of the proposed representational system depends on how well it can interface with visual input and motor ouput: no good doing away with co-ordinate representations in favour of sine waves, if it then becomes impossible for, say, a robot to determine the shapes of things it is looking at, or to adjust its movements in conformity with its beliefs about the spatial arrangement of its environment. These are therefore important questions for future work in this area. Other pending questions concern the extension of the proposed system to deal with non-convex shapes, 3-dimensional space, uncertainty and moving scenes. Finally, of course, many implementation details, some non-trivial, remain outstanding.

6 Conclusion

This paper has presented a system for detailed representation of convex plane figures. Its utility derives from the facility it provides to rotate and translate such regions, and to detect overlaps between them. Work remains to be done to extend and implement the system as proposed here. However, the results presented above suggest that, for detailed spatial representations, traditional co-ordinate systems may be considerably less efficient than the current approach.

References

[1] Brooks, R.A: "Symbolic Reasoning among 3-D Models and 2-D images", *Artificial Intelligence* **17**, pp.285-348 (1981).

[2] Klinger, A. and M.L.Rhodes: "Organization and Access of Image Data by Areas", *IEEE Trans. Patt. Anal. Mach. Intell*, PAMI-1 pp.50–60 (1979).

[3] McDermott, D.V. and E.Davis: "Planning Routes through Uncertain Territory", *Artificial Intelligence* **22**, pp.107–156 (1984).

An Unsupervised PDP Learning Model for Action Planning

Yoshiro Miyata
Bell Communications Research

Whenever we have in mind something that we wish to bring about in the environment, we must find an appropriate sequence of actions that will result in the desired state. For example, if we decide to move from one place to another, we need to find an appropriate path between the two places and an appropriate sequence of actions. Likewise, when we reach for an object we must find an arm configuration such that the arm's tip touches the object as well as a set of muscle contractions that will result in that configuration. These are problems we encounter and solve with ease hundreds of times every day. What is the mechanism that enables us to map from the representation of a goal to the representation of the action plan for realizing that goal, and how can such capability be learned from experience?

Much work has focussed on *execution* of actions, namely, how a representation of action is converted into the right sequence of actions (for example, Rumelhart & Norman, 1982; Mackay, 1982; Rosenbaum, Hindorff & Munro, 1987). Not much work has been done to understand *planning* of actions, namely, how to find an appropriate action sequence to achieve an environmental state. Works in this domain have tended to require hand-wiring of task specific structures into the system and thus are not readily applicable to more general situations (for example, Hinton & Smolensky 1984; Anzai, 1984). Recent development of learning algorithms for Parallel Distributed Processing (PDP) networks (Rumelhart & McClelland, 1986), especially the back-propagation (BP) algorithm (Rumelhart, Hinton, & Williams, 1986), enables a network to learn task specific structures based on general principles. There have been extentions of the BP algorithm to sequential action execution (Jordan 1986; Miyata 1987) which have shown that the networks exhibit a number of characteristics observed in human actions (Miyata 1988). However, as discussed below, planning of actions requires more than learning a single input/output mapping, which the BP algorithm is designed to do.

THE COMPUTATIONAL REQUIREMENTS

Figure 1 illustrates the computational requirements of the situations that are considered in this paper. The process starts from $E_{desired}$, a representation of some desired environmental state. The system generates a representation of an action plan, A_{plan}, which is then executed (A). One requirement is that there is no teacher that gives the system the desired actions. The feedback provided to the system for

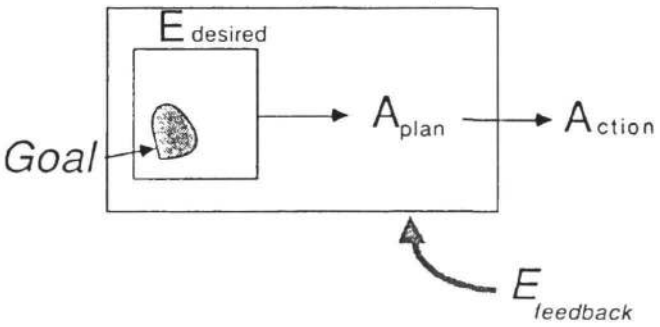


Figure 1. The computational requirements for the framework: (1) Feedback for learning is the environmental state as the result of an action plan; (2) A goal only partially specifies the desired environmental state. These requirements are not easily handled by the single mapping learning scheme.

learning is the environmental state as the result of executing the actions, $E_{feedback}$ (the *environment as the teacher* requirement). Another requirement is that $E_{desired}$ may be only partially specified by a goal of the task. For example, in a ship navigation task, the desired trajectory is usually only partially constrained by, say a gate or a channel (the *goal as partial specification of environment* requirement). Stated more precisely, a single goal specifies only some of the dimensions necessary to represent all possible goals. People seem to be able to learn and plan actions in situations defined by these requirements (see Anzai, 1984).

A common learning scheme is to characterize a task as a mapping from one representational space to another, and to train a network by presenting pairs of vectors from the input and the output space. One difficulty with applying this scheme to action planning is that the feedback provided to the learner from the environment does not necessarily specify what actions (outputs) the learner should have produced. Another difficulty is that, in order to generate an output, the input to a mapping must always be specified completely. This conflicts with the *goal as partial specification of environment* requirement. This paper presents one approach to these problems and proposes a framework in which the task-specific structure is learned as multiple mappings and planning is accomplished via interaction of these mappings that incorporates a constraint satisfaction process.

THE EAE NETWORK

The basic structure of the network, called the EAE (Environment-Action-Environment) network, is shown in Figure 2-(A). There is a set of units, called the E-units, for representing an environmental state, and another set, called the A-units, for representing an action plan. There are two mappings in the network: $E \rightarrow A$ mapping is the mapping from the E-units to the A-units; and $A \rightarrow E$ mapping is the mapping from the A-units to the E-units. Each mapping is implemented through a layer of hidden units.

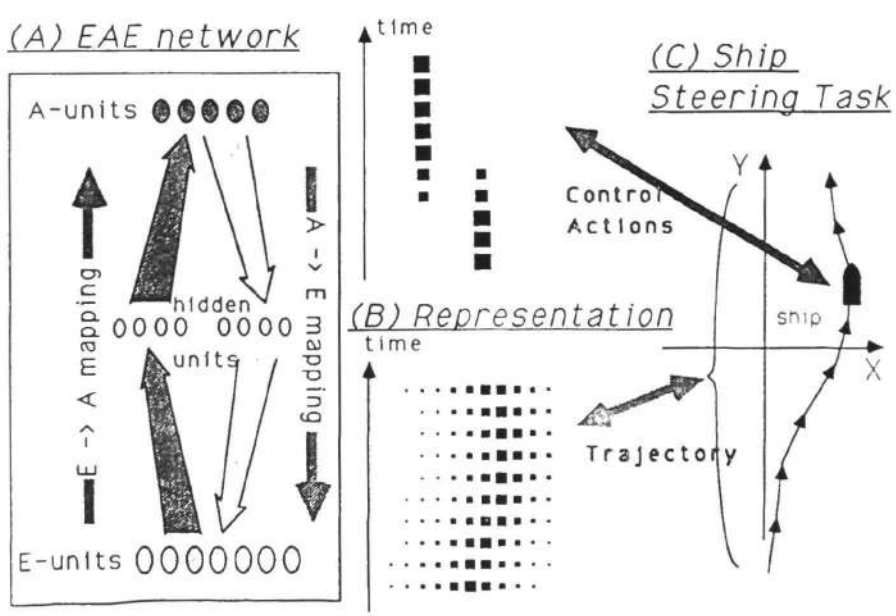


Figure 2. (A) The basic structure of the EAE network. E-units represent an environmental state and A-units an action plan. The $E \rightarrow A$ mapping maps from the E-units to the A-units, and the $A \rightarrow E$ mapping from the A-units to the E-units, each through a set of hidden units. (C) The network was applied to a ship steering task. The ship moves at a constant speed along the Y-axis of a 2-dimensional space and the network controls the acceleration of the ship along the X-axis. (B) The two matrices of squares show an activation pattern in the E-units representating a trajectory of the ship and an activation pattern in the A-units representing an action plan.

The EAE network has been applied to a task of navigating a ship in a 2-dimensional space (Figure 2-(C)), similar to an experimental task used by Anzai (1984). The ship moves at a constant speed along the Y-axis, and the network's action controls the acceleration along the X-axis. The E-units represent a trajectory of the ship, and the A-units a sequence of actions for controlling the ship. A trial consists of 10 time steps. A trajectory for a trial, shown on the right, is represented as the activation pattern in the (110) E-units shown as an 10×11 matrix of squares (Firuge 2-(B)). Each row of 11 units represents the X-position of the ship at a point in time by a coarse coding. The pattern in the (20) A-units, shown as an 10×2 matrix, represents an action plan for a trial. A pair of units in each row represents the control value at a point in time. The ship's acceleration to the right is proportional to the activation of the right unit minus the activation of the left unit. In addition, there is a set of units (not shown in the figure) for representing the context, i.e., the ship's initial position and velocity. These units are fully connected to both hidden layers.

The network learns by executing many quasi-random actions and observing what trajectories are generated. The $E \rightarrow A$ mapping receives the trajectory as the input and uses the action as the target. (In other words, it learns the knowledge of the form "to get this trajectory do this action".) The $A \rightarrow E$ mapping learns using the same action/trajectory pair, but the action as the input, and the trajectory as the target. ("If I do this action then this trajectory results.") Both mappings are learned only by interacting with the environment.

After the mappings in the network were trained to some criterion, the network was given various kinds of goals and it was able to find an appropriate action plan for each of the goals. The planning process starts from a specification of a goal. A goal in this task is some part of the trajectory that the ship must follow, such as a gate or a channel. Figure 3 illustrates an example goal which is a channel that the ship must be steered into. This goal is specified by the activation pattern shown in the middle, in which the units representing a part of the trajectory are given activation values representing the goal. The activation of other E-units are not constrained by the goal, and are given some default initial value, in this case, all zeros. When given such a goal, the network tries to fill in the unspecified part of the trajectory as well as an action sequence that will result in that trajectory. The planning process is achieved by the following steps (see Figure 4):

1 A goal is set by clamping some E-units and zero activation in other units.
2 From this pattern, the $E \rightarrow A$ mapping generates an action plan in the A-units. If a complete trajectory is specified in the E-units, this mapping can generate an appropriate action plan to achieve the trajectory. However, because the trajectory is only partially specified, the generated plan is unlikely to be appropriate.

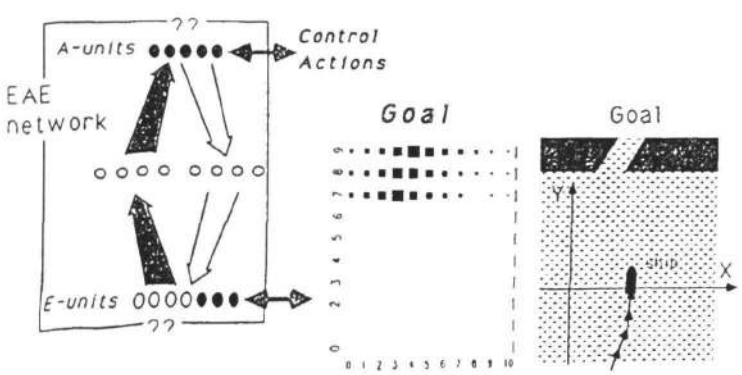


Figure 3. A goal is given by specifying activation values of some E-units representing partial trajectory, and the network tries to find activation patterns in the other E-units representing the rest of the trajectory, as well as an activation pattern in the A-units representing an action plan for achieving the trajectory.

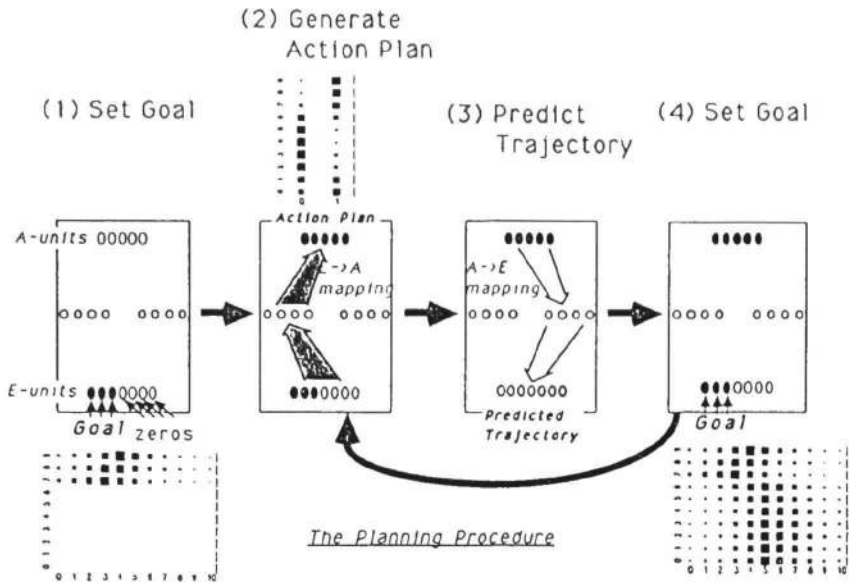


Figure 4. After the goal is specified in the E-units, the network maps back and forth between the E-units and A-units until the patterns settle into a stable state.

3 The $A \rightarrow E$ mapping generates, from the action plan, a predicted trajectory as the result of executing the plan.
4 This predicted trajectory may not satisfy the constraint of the goal. So, the E-units representing the goal are given the original values again.
5 The steps 2, 3, and 4 are repeated until the patterns settle into a stable state.

Figure 5 shows the series of patterns in the network during the process of planning after 1, 3, and 5 cycles. The leftmost column (**E**) shows the patterns in the E-units, after each time the goal is set. The second column (**A**) shows the patterns in the A-units. The third column (**Action**) shows the actual control values represented by these patterns. The rightmost column (**Trajectory**) shows what the trajectories would have looked like, if the actions were actually executed. As can be seen, the network was able to gradually improve the action plan. In this example, the patterns were stable after 5 cycles. This is one of

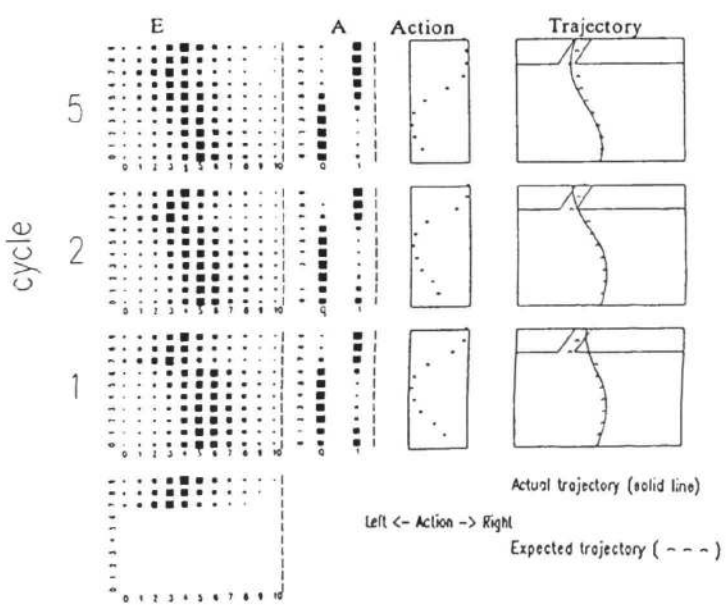


Figure 5. The activation patterns in the E-units (**E**) and the A-units (**A**) during the process of planning, after 1, 3, and 5 cycles. These patterns represent the trajectories and actions shown (**Trajectory**, and **Action**). The process starts from the representation of the goal in the E-units (bottom left) and the patterns in the E-units and in the A-units are iteratively mapped to each other until they are stable (cycle 5).

the most difficult cases, because a slight adjustment in the final direction of the ship requires a large change in the initial portion of the trajectory. Figure 6 shows the trajectories found by the network for 9 different goals and 3 different initial conditions.

ANALYSIS

Next, I present an analysis of the planning procedure and show that, under a number of assumptions, the process is expected to find an appropriate action plan. First, I make four assumptions. In figure 7, E-space is the space of all the possible activation patterns that can occur in the E-units. A-space is the space of all the possible activation patterns that can occur in the A-units. The first assumption is that all points in A-space are possible, i.e., they correspond to actual physical actions. This is reasonable because these patterns *cause* the physical actions. The second assumption is that only some points in E-space represent possible physical environmental states. This is reasonable because some physical states cannot be achieved by any action. Furthermore, some patterns may not correspond to any physical state. The third assumption is that the system learns the two mappings perfectly. The fourth assumption is that these mappings generalize to new patterns based on similarities to the learned patterns.

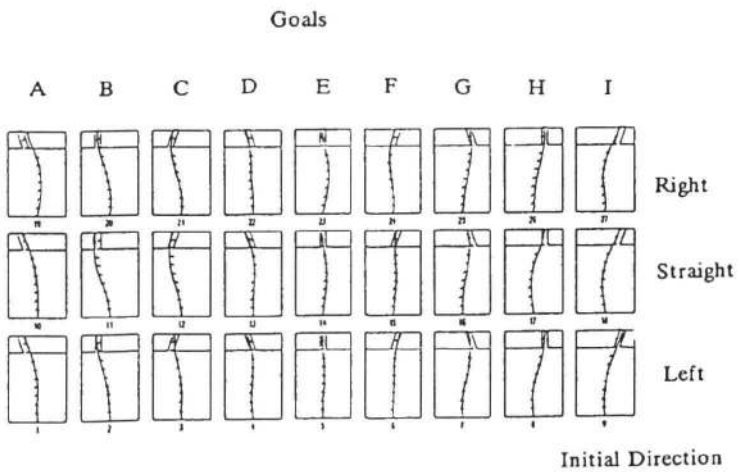


Figure 6. The trajectories found by the network for 9 different goals (A, B, .., I) and 3 different initial conditions (Right, Straight, Left).

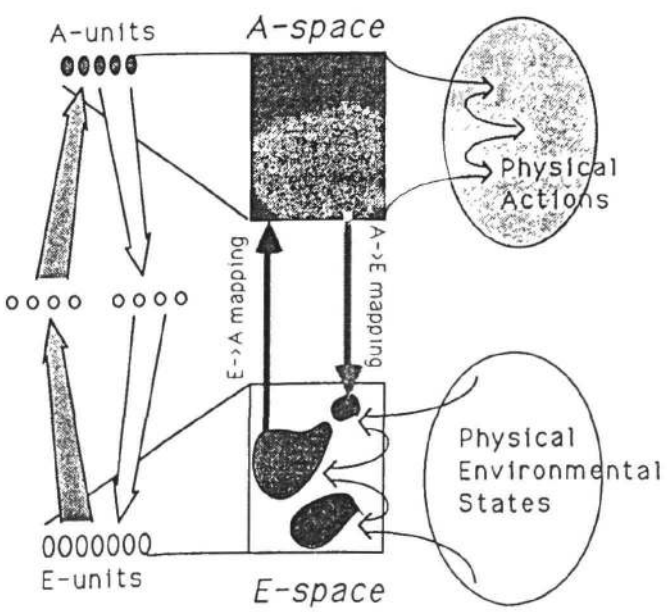


Figure 7. The four assumptions for the planning process: (1) All points in A-space represent some physical actions; (2) Not all points in E-space represent physical states; (3) Perfect $E \rightarrow A$ and $A \rightarrow E$ mappings; (4) Generalization based on similarity.

Such property has been demonstrated in many PDP networks (e.g., Cottrell, Munro, & Zipser, 1987; Chauvin, 1987).

Under these assumptions, the planning process can be analyzed as follows. (See Figure 8.) In E-space, the E_p regions are the points that represent possible physical states. For any particular goal, some dimensions of E-space are constrained by the goal. These dimensions, represented by the horizontal axis in the figure, correspond to the E-units that are given specific values by the goal. The other dimensions are not constrained by the goal. Thus, the goal defines a hyperplane in the space, called the E_c plane, which is parpendicular to all the constrained dimensions. The task of the planning is to find a *possible* state, a state that can be achieved by some action, that also satisfies the constraint of the goal. Such a state is represented by a point in an E_p region that is also on the E_c plane.

The EAE network searches for such a point by mapping back and forth between the two spaces. The network starts from $e_c^{(0)}$, a point in the E_c plane with zero activation values for the unconstrained dimensions. This point is mapped by $E \rightarrow A$ mapping to a point in A-space, $a^{(0)}$, and then mapped back by $A \rightarrow E$ mapping to a point in E-space, $e_p^{(1)}$. This point must fall within an E_p region because all points in A-space are possible and must be mapped to a point representing a possible state. This point may no longer on the E_c plane, and so the goal constraint is imposed again by projecting onto the E_c plane. This corresponds to clamping of some E-units to the goal. Using this point as the new starting point, the process is repeated. If it finds a solution after some iteration, the point no longer moves because a point in the E_p regions is mapped to itself. From the assumption that the mappings generalize based on similarity between patterns, or in this case similarities defined as the distance between points, it can be shown that, the distance between the points on the E_c plane ($e_c^{(0)}$, $e_c^{(1)}$, $e_c^{(2)}$, $\cdots$) and the points in the E_p regions ($e_p^{(1)}$, $e_p^{(2)}$, ..) will keep decreasing. Thus, the network will either find a solution or fall into a local minimum, depending on the shape of the E_p regions in relation to the E_c plane.

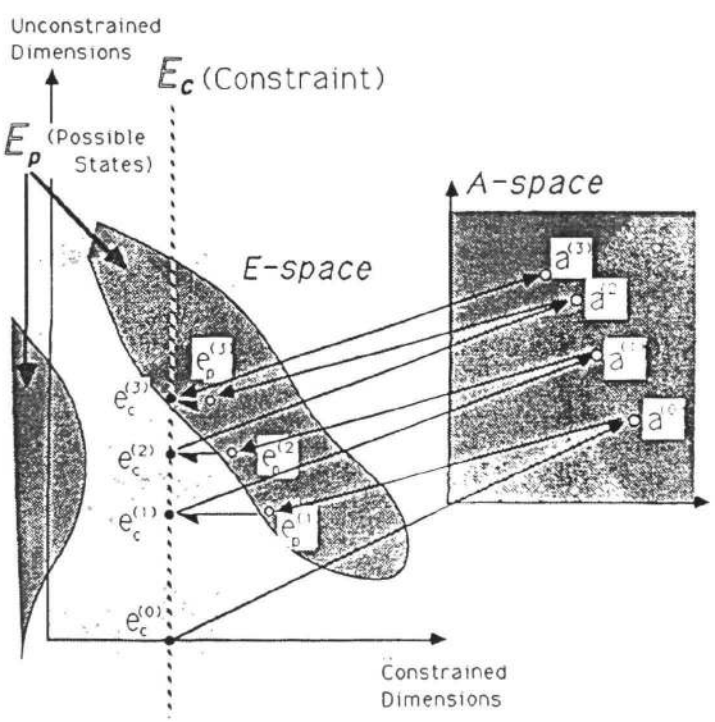


Figure 8. An analysis of the planning process. A solution is a point in E-space that both is possible (in an E_p region) and satisfies the goal (on the E_c plane). The network searches for such a point by mapping back and forth between E-space and A-space.

SUMMARY AND EXTENSION

The EAE network gives one explanation of how learning and planning of actions can be accomplished, using only feedback from the environment and when the desired state of the environment is only partially specified. Obviously, in order to evaluate the framework it is necessary to test on many different tasks and also to compare the performance of the network more closely to that of humans. The advantage of the EAE framework is that it does not need any domain specific structure, except a design of representation of the environment and of the actions, and thus it is readily applicable and testable in other domains. Furthermore, the framework can be extended in several interesting ways. First, it is obvious that people can modify a planned sequence of actions based on feedback from the environment, especially when the action is slow. The system could use feedback from the environment during the execution of an action plan to adjust its prediction of future states and the action plan. This would allow accurate performance without perfect prediction and therefore without perfect learning of the mappings. Second, the framework provides a possible way to model automatization of a skill when actions and environmental contexts are sufficiently correlated (Shiffrin & Schneider, 1977). In such situations, the system could establish a mapping from the contexts to actions, which could speed up, or possibly eliminate the need for, the process of generating predictions and comparing with explicit repesentations of goals.

REFERENCES

Anzai, Y. (1984). Cognitive control of real-time event-driven systems. *Cognitive Science, 8*, 221-254.

Chauvin, Y. (1987). *Generalization as a function of the number of hidden units*. Unpublished manuscript.

Cottrell, G. W., Munro, P., & Zipser, D. (1987). Image compression by back propagation: an example of extensional programming. In N. E. Sharkey (Ed.), *Review of Cognitive Science I*. Norwood, NJ: Ablex.

Hinton, G. E., & Smolensky, P. (1984). *Parallel computation and the mass-spring model of motor control*. Report 123, Center for Human Information Processing, UC San Diego.

Jordan, M. I. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. In *Proceedings of the eighth annual conference of the Cognitive Science Society* (pp. 531-546). Amherst, MA.

MacKay, D. G. (1982). The problems of flexibility, fluency, and speed-accuracy trade-off in skilled behavior. *Psychological Review, 89*, 483-506.

Miyata, Y. (1987). Organization of action sequences in motor learning: a connectionist approach. In *Proceedings of the ninth annual conference of the Cognitive Science Society* (pp. 496-507). Seattle, WA.

Miyata, Y. (1988). *The learning and planning of actions*. PhD thesis, Psychology Department, UC San Diego. Tech. Rep. No. 8802, Institute for Cognitive Science, UC San Diego.

Rosenbaum, D. A., Hindorff, V., & Munro, M. (1987). Scheduling and programming of rapid finger sequences: tests and elaborations of the hierarchical editor model. *Journal of Experimental Psychology: Human Perception and Performance, 13*, 193-203.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representation by error propagation. In D. E. Rumelhart, J. L. McClelland (Ed.), *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations*. Cambridge, MA: MIT Press/Bradford Books.

Rumelhart, D. E., & McClelland J. L. (1986), *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations*. Cambridge, MA: MIT Press/Bradford Books.

Rumelhart, D. E., & Norman, D. A. (1982). Simulating a skilled typist: A study of skilled cognitive-motor performance. *Cognitive Science, 6*, 1-36.

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.

Representation and Recognition of Biological Motion [2]

Nigel H. Goddard [1]
Hughes Artificial Intelligence Center
23901 Calabasas Road Calabasas, CA 91302
and
Division of Biology
California Institute of Technology

I. Introduction

The human visual system has a remarkable ability to discriminate between different types of movement. The classic illustration of this ability is Johannson's Moving Light Display (MLD) [Johansson, 1973]. Reflective pads were placed at the joints of an actor dressed in black, and the actor illuminated. Films were taken of the actor walking, jumping and making various other movements against a black backdrop. When these films were shown to subjects, they all recognized the display to be of a person walking, jumping, *etc.*, but reported single frames to be meaningless patterns of dots. A presentation time of no more than 200 msec was sufficient for all subjects to make the correct discrimination. In forced choice experiments, all subjects accurately identified 6 human and 3 puppet generated patterns with a presentation time of 400 msec. Further experiments [Kozlowski and Cutting, 1977, Cutting and Kozlowski, 1977] demonstrated the sensitivity of this faculty: subjects could determine the actor's gender, and could even identify the actor if (s)he was known to the subject.

This paper describes the early stages of an attempt to produce a computational account of this capability, consistent with the psychological and neurophysiological literature. The following section discusses data on the human visual system which may shed light on MLD processing. Section 3 introduces Feldman's Four Frames computational architecture [Feldman, 1985] for the visual system, outlines the low level processing we believe occurs, and develops and motivates our target representation the *scenario*. Section 4 describes the processing architecture which activates scenarios using the output of the low level system, and details its implementation in a connectionist network.

II. MLD Processing in the Human Visual System

There are two obvious ways the motion information available in the Johannson experiments could be used to generate the percepts of *person* and *walking*, or the single percept of *walking person*. The first method would be to use the motion information to index directly into memory, implying a memory representation rich in temporal information. This method places motion information in a central position *vis-a-vis* the recognition process. The second method would use the motion information to reconstruct various static qualities of the scene object (such as structure), and use those static qualities to index into memory and recognize the object. Having recognized the object, the motion of various key parts of the object could be used to discriminate between gaits. In this second method, the motion information is used in two ways: to recover static qualities; and to disambiguate a small number of gaits.

In this paper we address the first method, but do not rule out the second. Such a motion-specific process and memory structure must play a role in MLD experiments. Johansson's subjects could distinguish gait with a presentation time of less than a quarter of a cycle of the periodic motion (i.e. less than a quarter of a step in walking or running) [Johansson, 1976]. This implies that phasal relationships between joints throughout a cycle of the gait must be represented in memory. Recognition must be based on invariants and the absolute dot motions in an MLD are not invariant with respect to scale or rotation in the image plane. Something like the Johannson's Visual Vector Analysis [Johansson, 1973] must be taking place, with the movement of dots treated as relative to that of other dots. Relative speed of rotation about a joint of the limbs connected at the joint is invariant with respect to scale and rotation in the image plane, and may be a good candidate for the recognition process. However the fact that upside-down MLDs are not recognized as such [Sumi, 1984], while upside down moving stick figures are easily recognized (our own informal observation) implies that the motion invariants used in recognition cannot be computed by the visual system for upside down MLDs. This implicates top-down feedback in the computation of the invariants, under the assumption that the gait is represented in memory for the object in its normal orientation. The memory representation then provides no help in computation of invariants for moving objects in an unfamiliar orientation.

Further evidence for memory structures devoted to representation of sequence and time is provide in [Freyd, 1983]. She presented single frames of a motion sequence to subjects, and then tested their memory for other frames

[1] correspondence should be sent to the first address

[2] An early version of this paper is to appear in *Proceedings of DARPA Image Understanding Workshop 1988* under the title "Recognizing Animal Motion".

from the same sequence. Subjects found it harder to distinguish frames *later* in the sequence from the stimulus frame than they did to distinguish frames *earlier* in the sequence from the stimulus frame. [O'Connell and Gerard, 1985] found that children develop the ability to reproduce familiar sequences earlier than the ability to reproduce the same events prevented in an unfamiliar sequence, implying an early development of representation of sequence. [Runeson and Frykholm, 1983] argue persuasively that representation of body motion is couched in terms of causal factors as well as descriptive components (i.e. force and mass as well as velocity). MLD's of actors lifting boxes were presented to subjects who had no difficulty discriminating the box's weight qualitatively. If the actor attempted to deceive, not only was the deception detected but both real and intended weights were discriminable. Although we will ignore causaul factors in our model, they will have to be integrated eventually.

Whatever the interpretation of the psychological results, it is clear there must be a memory structure rich in information about change in the environment. Brain damaged patients provide neuroanatomical evidence for a separate recognition process based on motion alone. Lesions to the temporal lobe can lead to the inability to identify faces, while leaving intact the ability to identify from body motion; and lesions to parietal cortex can impair recognition from body motion while leaving object recognition unimpaired [Damasio, 1988]. [Perret *et al.*, 1985] found cells in the superior temporal sulcus of the macaque monkey which responded selectively to differing body motions in view, into view and out of view. [Chitty *et al.*, 1987 (in submission)] found cells in the same area that responded selectively to MLD displays, to MLD displays with limb segments suggested by contour; also some cells selective for static form which responded to MLD stimuli, implying computation of form from motion. All these results indicate representation and recognition of sequence is explicitly performed in the visual system.

III. Four Frames

We take as a basis for our computational model the Four Frames architecture [Feldman, 1985] for the visual system, and its extension to deal with kinematics [Feldman, 1988]. Using this computational framework, we may address the question of how MLD image sequences are processed into percepts. Of the four frames (*retinotopic, stable-feature, world-knowledge* and *environmental*), we shall focus on the interaction between the stable-feature frame and the world-knowledge formulary.

A. Retinotopic to Stable-Feature

The first frame, the *retinotopic frame*, is "intended to model the view of the world that changes with each eye movement" [Feldman, 1985, page 265]. In our case the information in the retinotopic frame at any instant is a representation of the dot pattern image on the retina at that instant, modulated by the receptor properties and their time characteristics. The most salient information here will be retinal smear due to the motion of the dots. The second frame, the *stable-feature frame*, computes intrinsic features of the scene being viewed, which do not change with eye-movement. For our purposes, the first important feature encoded in the stable-feature frame will be the motion of the dots. Whether via the short range process or apparent motion, the stable feature frame will provide at each instant the velocity and position of each dot in the moving pattern. Considerable work has been done on the detailing the kind of computations necessary for transformating motion information from the retinotopic to stable-feature frame [Hildreth, 1983]. We assume this computation is performed along the lines suggested in [Feldman, 1988] and [Olson, 1988].

However, as stated in the previous section, the position and motion parameters for each dot in the MLD images are not suitable data to use in indexing. The stable-feature frame must also compute the invariants in terms of which object motion is represented in the next frame, the world-knowledge formulary. We suggested above that for biological motion relative speed of rotation of the two limb segments about their common joint might be such an invariant. In fact we also need to know direction of rotation, and will need some relative positional information, such as angle formed at the joint. In the following subsection we develop a representation for biological motion, the *scenario*, based on relative angular velocity and relative angular position of limb segments. We assume that the invariants used to represent scenarios are computed in the stable feature frame from the position and velocity parameters of the MLD dots, using top-down feedback. The nature of the feedback and the details of the computation are hard unsolved problems, but we would assume at least that the feedback is from the same type of memory used to hold scenarios. The recognition problem is then to index into scenario memory from the invariant values computed in the stable-feature frame.

B. Representing Biological Motion in the World-Knowledge Formulary

Under the Four Frames analysis, scenario memory is contained in the *world-knowledge formulary*. This frame is "the observer's general knowledge of the world, including items not dealing with either vision or space" [Feldman, 1985, page 266]. Knowledge of types of movements of types of objects is general knowledge of the world.

Our scenario representation will be couched in terms of visual *events*. Informally an event is any significant change in one of the invariants specifying the object and motion. A *form event* could be arrival at colinearity of various object features, a *color event* a change in color, and a *motion event* a change in speed or direction of movement.

To make our task tractable, we make the following assumptions. The motion to be recognized is that of an articulated stick figure with bright spots at the joints, moving

parallel to the image plane and viewed orthogonally. If the trunk of the figure is moving, we assume the imaging system is tracking the center of rotation of the trunk, so that in the image the trunk is undergoing pure rotation. The limbs are rotating about an end of the trunk, and so on. Thus we have a movement which can be completely described by the length of each stick in the figure, and the change in angle at each joint over time. We shall assume that the change in joint angle over time is piecewise linear, i.e. a sequence of segments of constant angular velocity. This treatment is similar to that of [Johansson, 1973] and recalls Cutting's hierarchy of "centers of moment" [Cutting, 1981]. As in the related Tinker Toy recognition project [Cooper and Hollbach, 1987], we have assumed a principal views treatment and features which are invariant to scale[3]. As viewpoint changes, so that motion is no longer parallel to the image plane, these angular position and velocity cues vary little and can be considered invariant for the purposes of indexing.

We have now delimited the class of movements in such a way that a complete representation is possible. Each joint undergoes a sequence of constant angular velocity changes, which for biological movements such as walking is periodic. The set of sequences *together with information co-ordinating them* describe the motion completely: sufficiently to unambiguously regenerate it. Such a set of sequences of events constitute a *scenario*.

The fundamental motion *event* under these assumptions is a change in angular velocity. The choice of angular velocity as the basis of motion representation is due to data suggesting that we have velocity information availible for a variety of tasks, but not any higher derivative such as acceleration [Jagacinski *et al.*, 1983, Todd, 1981, Runeson, 1975]. There are cells sensitive to rotation [Saito *et al.*, 1986, Sakata *et al.*, 1985], although not sufficiently highly tuned for particular velocities. However the output of several broadly tuned cells can be combined to achieve finer tuning. [Perret *et al.*, 1985] found cells sensitive to velocity change, which would be required to detect our events. But these neurophysiological data are more an indication of what is possible in the visual system rather than definitive evidence for a particular computational or representational scheme.

A simple graphical representation follows from the specification of a scenario given above. We represent each event, or point (in time) when the angular velocity changes, by a graph node. The nodes are labeled with the new angular velocity and the absolute angle of the joint at that time. Directed edges between nodes represent sequence, each edge being labeled with the time between the two nodes. Each sequence is represented by such a graph. The graph is cyclical if the sequence is periodic. The graphs for the sequences are linked with directed edges that specify the *co-ordination* between the sequences, using labels on the edges as before.

[3]In fact cells selective for faces by principle view have been found in the macaque monkey [Perret *et al.*, 1987].

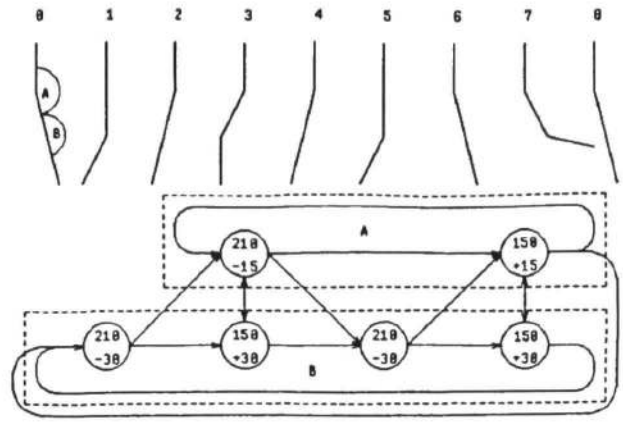


Figure 1: **abstract scenario and graph**

Figure 1 shows an example abstract scenario and its associated graph. At the top of the figure we show two connected pendula, rotating at joints A and B. The topmost stick is stationary. Over the course of eight time steps, the pendula undergo the motion depicted from left to right. Inter-step motion is of constant angular velocity. Thus it is easily seen that the upper pendulum (A) oscillates with period 8, and the lower pendulum (B) oscillates with period 4. For each pendulum, a cycle of this artificial oscillation consists of two constant angular velocity segments, one clockwise and one counter-clockwise. In the graph we represent changes of angular velocity by nodes. Thus there are two nodes for joint A, shown in the upper dotted box, corresponding to the two changes of direction of rotation during the 8 step cycle. Similarly for joint B there are four nodes corresponding to the four changes of direction during two cycles of the 4 step period. The nodes are labeled with the angle at the joint and the new angular velocity at the joint. For example, the leftmost node in the dotted box for A specifies that this event occurs when the joint angle is 210 degrees and the new angular velocity is -15 degrees/step. Within each dotted box are the sequence links indicating the order of the nodes. Links between the dotted boxes indicate coordination of the joint sequences. The critical *simulataneity* links are shown by the two lines with arrows at both ends.

IV. Connectionist Network Implementation

The preceding section described the information availible in the stable-feature frame and the world-knowledge formulary. For our implementation, the stable-feature frame provides the input representation, and the world-knowledge formulary the memory representation. Recognition is the process of using the stable-feature frame information to activate structures in the world-knowledge formulary.

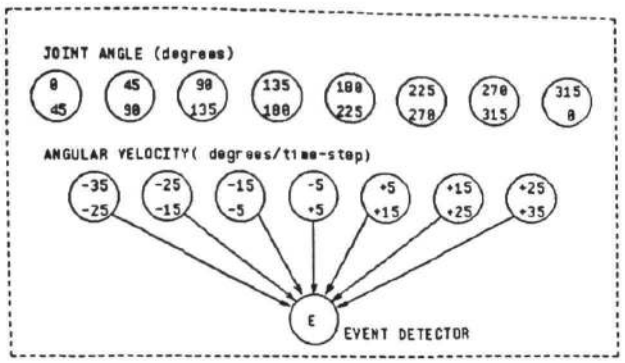


Figure 2: **input module**

A. Input and Scenario Representation

The stable-feature frame will compute the invariants used for recognition. The invariants we have chosen are the angular position and velocity at each joint. Using the unit value principle [Ballard, 1986] we have as input representation a number of *input modules*, each providing the angle and angular velocity for one joint.

Figure 2 shows an input module. Each input module consists of a set of *angle*, a set of *angular velocity* units, and a single unit to detect events (change in angular velocity). Each angle and velocity unit responds to a range of values, shown by the two numbers in each unit; for example, the first angle unit responds to joint angles from 0 to 45 degrees. This figure is an example of an input module for illustrative purposes. In the implementation, we actually use many more units to represent the 360 degree range. The pattern of activity of these units over time describes the kinematics of the joint. We assume that the stable feature frame has segmented the image information into information for each joint, and activates the appropriate number of input modules.

How should we represent scenarios? The graphical representation developed above is naturally implemented as a connectionist network[4]. Each graph node is represented by a unit, and each directed labeled edge by a link with an associated time-delay. The units have a site for *priming* activation which arrives along these delay links, and another site for input from the lower levels of the visual system. Initially all units receive a small amount of priming activation. Units expect activation to arrive at both sites simultaneously. If priming activation or input activation arrives, but not both, then the events in the image are not corresponding to the scenario represented. If the image events do indeed correspond to the scenario represented, then priming activation should flow through the network, building up as it does so. For periodic motions this activation should saturate quickly. Figure 1 is easily re-interpretable as a connectionist network, the graph

[4] In our networks units have one or more *sites* at which links arrive, and where input activation is processed. This enables differential treatment of inputs.

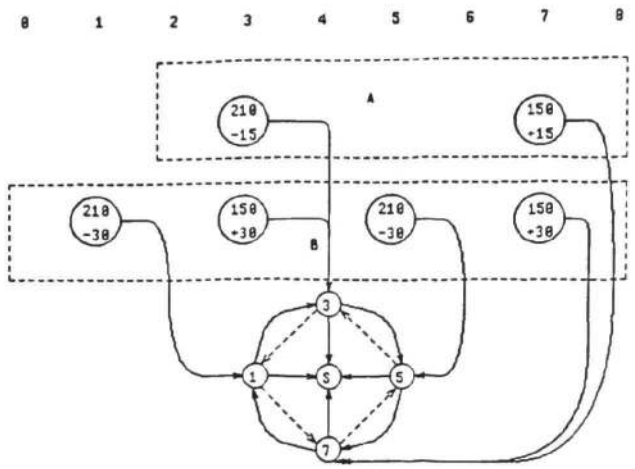


Figure 3: **evaluation network**

nodes becoming scenario event units.

These scenario event units are complex. The unit has four sites: one receives inhibition from other event units in the same sequence; one receives priming from event units that represent simultaneous events in other sequences; another receives priming from preceding events; and the last receives input originating in the input modules. The sites compute relatively complex functions of their inputs, such as sum of squares and exponential decay. The unit output is a multiplicative function of the total priming (less inhibition) and the input activation.

A scenario is recognized when activation flows around the network representing it. It is a simple matter to attach a network to the scenario network to detect when and how strongly activation is flowing through the scenario network. The output of this *evaluation* network is a measure of how similar the input is to this particular scenario. Figure 3 illustrates the evaluation network. The dotted boxes from Figure 1 are reproduced, together with the time step count (0 to 8). The evaluation network is the bundle of five units at the bottom of the diagram. The central unit is the summator, which computes the final evaluation. The four peripheral units represent the four time steps at which events take place for this scenario; they are labelled in the diagram with the time steps they represent. If the scenario event units are activated in the correct order, then they will activate these evaluation units in the correct order. The solid links represent the correct order of activation. These have an associated delay equal to the time step difference between the source and destination units. The dotted links are reverese direction inhibitory links, with no delay associated. The links ensure that the evaluation units will not become significantly active unless they are being activated in the correct order. The central summation unit checks that only one is active at a time.

B. Recognition

Assuming the input described above, i.e. at each time step, for each joint in the image, a readout of the angle and angular velocity at that joint, how do we index into scenario memory? We would like the indexing algorithm to be tolerant of missing data points (for example, due to occlusion), and to incrementally converge on the correct scenario as more and more data arrives. At the same time we must avoid exponential growth in the number of units and links required as the number of scenario memories increases. We would also like to be able to take advantage of evidence based on structural or other static qualities of the object if it is available.

Our input modules detect *changes* in angular velocity, thus discretizing the input into a set of sequences of events at which angular velocities change for each joint. These sequences are exactly analogous to the sequences of events represented by the nodes in the scenario graph. For a given scenario we must match the input sequences against the stored sequences to determine which input sequence corresponds to which stored joint sequence. Not only must a mapping from input to scenario be established, the *co-ordination* between input sequences must match the co-ordination between joint sequences in the scenario. We must perform this match for each scenario memory. If we assume a solution to the first problem (matching a particular scenario against the input), then we can achieve recognition time independent of the number of scenarios stored in memory at a cost of linear increase in the number of units and links: we match against all scenario memories in parallel. This is trivial to do in a connectionist network; we simply duplicate the matching machinery for each scenario.

C. The Correspondence Problem

Solving the correspondence problem is harder. We cannot wait until we have all the data before attempting to match. This must also be an incremental process over time. Our approach is to attempt to match all input sequences against all stored sequences in parallel. Again it is trivial to achieve parallel matching in a connectionist network, if one is willing to pay the price in terms of the number of units and links required.

Figure 4 gives a schematic outline of the functional architecture we adopt to solve the correspondence problem. This diagram shows four functional units, depicted by the bold boxes. The input modules and evaluation network, at the top of the diagram, were detailed above. In this figure we show three input modules labelled #X, #Y and #Z. On the left is an example scenario graph network, with three event sequences labelled #A, #B and #C. Each dotted box in the scenario graph represent the event sequence for one joint sequence, as in Figure 1. There is one scenario graph network for each scenario in memory. In the middle of Figure 4 is the grid of binding networks (the dotted boxes). There is one such grid for each scenario graph in

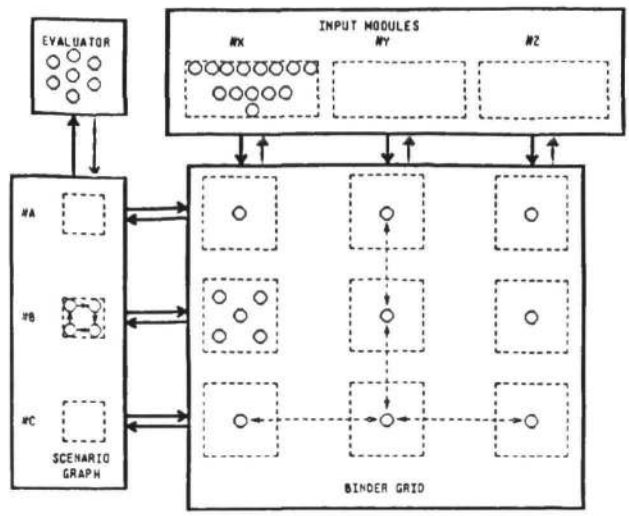


Figure 4: **functional architechture**

memory. A binder grid is always composed of n by n binding networks, where n is the number of event sequences in the scenario which the binder grid is associated with. The heavy arrows between the functional units indicate activation flow, the striped arrows indicating top-down feedback not yet implemented. Each input module sends activation to the column of binding networks below it in the diagram. Each row of binding networks sends activation to the scenario event sequence which is directly to its left, and receives feedback from that event sequence. The scenario event sequences send activation to the evaluation network as shown in Figure 3.

The function of the binder grid is to establish a one to one mapping, or correspondence, between the active input modules and the event sequences in the scenario graph. The input is a time-varying pattern of activation over the set of input modules. There is one input module for each joint in the scene. The the time-varying pattern at an input module should match the expected pattern represented in one of the scenario sequences. The binder grid compares in parallel the sequence of events at each input module with the events in each scenario event sequence. This is achieved by having a separate binding network for each input-module/scenario-sequence pair. Thus the top left dotted box in the binder grid in Figure 4 represents the binding network that is attempting to match the events arriving at input module #X with the events represented in scenario sequence #A. Binding networks with competing interpretations are arranged to inhibit each other, so that if a consistent interpretation can be found there will be exactly one binding network active in each row and column of the grid.

If the match between the input events is close to the sequences in a scenario, a good binding will be found, activation will flow around the scenario network, and the evaluation network will become active. If there is a partial match, the activation in the scenario network will be

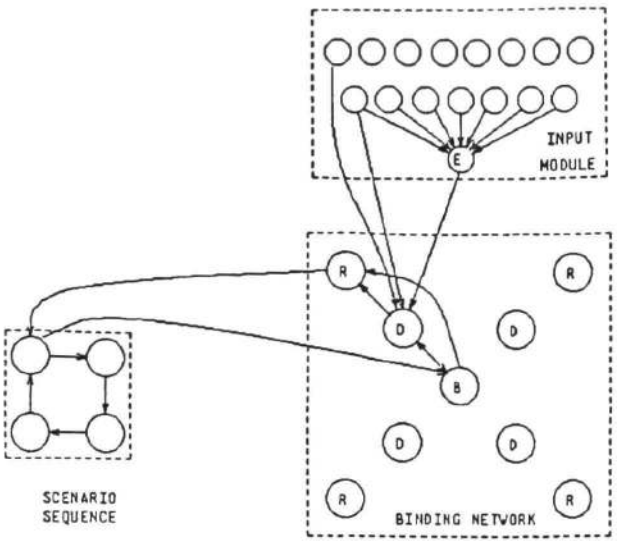


Figure 5: **binding network details**

lower, and less consistent in time, so that the evaluation network may become somewhat active. The best match is found by reading off the activity of the summator units in the evaluation network for each scenario.

Figure 5 shows the details for one binding network. At the top is an input module. It has a set of units tuned to a particular angular range and another set tuned to particular angular velocity. Unit E is an event detecting unit, connected to all the velocity tuned units in the module; it fires when the angular velocity changes, our definition of an event. At the left of Figure 5 is a sequence network from a scenario, similar to the dotted boxes in Figure 1. In the middle is a binding network.

A binding network performs two functions: it passes on input events from its input module to its sequence network; and it compares the events occurring in the input module with those occurring in the sequence network. Events are differentiated by the angle and the angular velocity at the joint, and occur when the angular velocity changes. For each event represented in the sequence network, there is a pair of units in the binding network, a *detector* unit, labeled D, and a *relay*, labeled R. In Figure 5 there are four such pairs, corresponding to the four event units in the scenario sequence. We show the links for one pair. The detector unit fires fires when the appropriate event occurs in the input module, i.e. when all three inputs from the input module are active. The relay unit passes on activation from the detector unit to the appropriate event unit in the scenario sequence network, modulated by the level of activation of the *binding* unit, labeled B. The detector unit also sends activation to the binding unit. This binding unit has a site for each event represented in the sequence network. Each site receives activation from a network event unit and from the corresponding detector unit. If a site receives input from the detector unit, it expects input soon after from the sequence network. Otherwise there is a mis-match occurring. The binding unit checks that the sites are receiving activation in this fashion, and if so increases its activation level. Otherwise its activation decreases.

All binding networks connected to the same input module have their binding units arranged in a mutually inhibitory network (see Figure 4). Similarly all binding networks connected to the same sequence network are arranged so that their binding units inhibit each other. Thus, even with locally ambiguous input, so long as globally the input is determinate, the correct scenario should be the most highly activated. If evidence for matching is available from other sources, for instance form or color matching, it can be used to influence the scenario match by providing input to the binding units in the binding networks.

D. Preliminary Qualitative Results

The architecture has been implemented using the Rochester Connectionist Simulator [Goddard *et al.*, 1988] for two scenarios - one abstract, and one corresponding to a running stick figure. Unsurprisingly, with such distinct choices, the network had no problem with discriminating inputs. The results depend on the actual parameters used in the activation functions and in the time-delayed links. As expected, presenting perfect input causes the scenario network to saturate quickly (within one cycle of the motion). Presenting imperfect input, e.g. with one of the input modules inactivated to simulate occlusion, caused the scenario network to activate more slowly. Recognition was fairly robust over quite large parameter variations. Overall it is clear that the architecture solves the problem, and moreover that it can be tuned along several dimensions: speed, sensitivity to missing data, sensitivity to incorrect data. Exactly how the network should be tuned is a matter for further research, and will require psychophysical experiments.

V. Conclusions

We have introduced a representation for articulated stick figure motion that is naturally implemented in a massively parallel network. A network architecture for indexing into this memory representation from biologically plausible input has been designed. The results of the preliminary implementation and tests are encouraging.

VI. Acknowledgements

Inspiration for this work came from Jerome Feldman at the Department of Computer Science, University of Rochester. Funding was generously provided by Hughes Aircraft Company. My colleagues at Hughes Artificial Intelligence Center critiqued a previous draft of this paper. Christof Koch at the California Institute of Technology, Division of Biology, gave financial support and help with the biological literature.

References

[Ballard, 1986] Dana H. Ballard. Cortical connections and parallel processing: structure and function. *The Behavioral and Brain Sciences*, 9(1):67–120, March 1986.

[Chitty *et al.*, 1987 (in submission)] Andrew J. Chitty, David I. Perret, Amanda J. Mistlin, and M. Harries. Visual cells sensitive to biological motion. 1987 (in submission).

[Cooper and Hollbach, 1987] Paul R. Cooper and Susan C. Hollbach. Parallel recognition of objects comprised of pure structure. In *Proceedings DARPA Image Understanding Workshop*, pages 381–391, February 1987.

[Cutting and Kozlowski, 1977] J.E. Cutting and L.T. Kozlowski. Recognizing friends by their walk: gait perception without familiarity cues. *Bull. Psychonometric Soc.*, 9(5):353–356, 1977.

[Cutting, 1981] James E. Cutting. Six tenets for event perception. *Cognition*, 10:71–78, 1981.

[Damasio, 1988] Antonio R. Damasio. Seminar talk, California Institute of Technology, February 1988.

[Feldman, 1985] Jerome A. Feldman. Four frames suffice: a provisional model of vision and space. *The Behavioral and Brain Sciences*, 8:265–289, 1985.

[Feldman, 1988] Jerome A. Feldman. *Time, Space and Form in Vision*. Technical Report, Department of Computer Science, University of Rochester, 1988.

[Freyd, 1983] Jennifer J. Freyd. The mental representation of movement when static stimuli are viewed. *Perception and Psychophysics*, 33(6):575–581, 1983.

[Goddard *et al.*, 1988] Nigel H. Goddard, Mark A. Fanty, and Kenton Lynne. *The Rochester Connectionist Simulator*. Technical Report TR213, Department of Computer Science, University of Rochester, 1988.

[Hildreth, 1983] Ellen Catherine Hildreth. *The measurement of visual motion*. MIT Press, Cambridge, MA, 1983.

[Jagacinski *et al.*, 1983] Richard J. Jagacinski, Walter W. Johnson, and Richard A. Miller. Quantifying the cognitive trajectories of extrapolated movements. *Journal of Experimental Psychology: Human Perception and Performance*, 9(1):34–57, 1983.

[Johansson, 1973] Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception and Psychophysics*, 14:201–211, 1973.

[Johansson, 1976] Gunnar Johansson. Spatio-temporal differentiationn and integration in visual motion perception. *Psychological Research*, 38:379–393, 1976.

[Kozlowski and Cutting, 1977] L.T. Kozlowski and J.E. Cutting. Recognizing the sex of walker from dynamic point-light displays. *Perception and Psychophysics*, 21(6):575–580, 1977.

[O'Connell and Gerard, 1985] Barbara G. O'Connell and Anthony B. Gerard. Scripts and scraps: the development of sequential understanding. *Child Development*, 56:671–681, 1985.

[Olson, 1988] Thomas J. Olson. *Forthcoming dissertation*. PhD thesis, University of Rochester, 1988.

[Perret *et al.*, 1985] D.I. Perret, P.A.J. Smith, A.J. Mistlin, A.J. Chitty, A.S. Head, D.D. Potter, R. Broennimann, A.D. Milner, and M.A. Jeeves. Visual analysis of body movements by neurones in the temporal cortex of the macaque monkey: a preliminary report. *Behavioural Brain Research*, 16:153–170, 1985.

[Perret *et al.*, 1987] David I. Perret, Amanda J. Mistlin, and Andrew J. Chitty. Visual neurones responsive to faces. *Trends in Neuroscience*, 10(9):358–364, 1987.

[Runeson, 1975] Sverker Runeson. Visual prediction of collision with natural and nonnatural motion functions. *Perception and Psychophysics*, 18(4):261–266, 1975.

[Runeson and Frykholm, 1983] Sverker Runeson and Gunilla Frykholm. Kinematic specificattion of dynamics as an informational basis for person-an-action perception: expectation, gender recognition, and deceptive intention. *Journal of Experimental Psychology: General*, 112(4):585–615, 1983.

[Saito *et al.*, 1986] Hide-aki Saito, Masao Yukie, Keiji Tanaka, Kazuo Hikosaka, Yoshiro Fukada, and Eiichi Iwai. Integration of direction signals of image motion in the superior temporal sulcus of the macaque monkey. *Journal of Neuroscience*, 6(1):145–157, 1986.

[Sakata *et al.*, 1985] H. Sakata, H. Shibutani, K. Kawano, and T. L. Harrington. Neural mechanisms of space vision in the parietal cortex of the monkey. *Vision Research*, 25:453–463, 1985.

[Sumi, 1984] Shigemasa Sumi. Upside-down presentation of the johansson moving light-spot pattern. *Perception*, 13:283–286, 1984.

[Todd, 1981] James T. Todd. Visual information about moving objects. *Journal of Experimental Psychology: Human Perception and Performance*, 7(4):795–810, 1981.

When half right is not half bad: Hypothesis testing under conditions of uncertainty and complexity

Joshua Klayman

Center for Decision Research
Graduate School of Business, University of Chicago

Analyses of scientific reasoning, from computer simulation (e.g., Langley et al., 1987) to biographical analyses of famous scientists (e.g., Tweney, 1985), often rely on a prototypical model of the hard sciences, especially physics. On the basis of this prototype, scientific inquiry has usually been modeled in terms of the discovery of laws, like Newton's, Boyle's or Ohm's--the kinds of simple formulae learned in introductory science classes. Although simple, each law explains a broad class of events or relations. These laws have exceptions and complications, especially in exotic conditions, but basically, F = ma for all kinds of masses and all kinds of forces, and PV = nRT for all kinds of gasses in all kinds of containers. The laws are, in principle, deterministically correct, and, within the bounds of measurement error, the data about them are consistent and unambiguous.

Unfortunately, this model of science is not a good representation of hypothesis testing and scientific reasoning in informal settings. In general, the subjects of everyday reasoning (e.g., the behavior of children, the performance of automobiles, the judgments of editors) are not subject to simple explanations. In a lawful science, even slight discrepancies from the law are matters to be reckoned with. With intuitive theories, half right is not half bad: One is often pleased with an explanatory value noticeably better than zero, and the domain of applicability of such theories is usually quite restricted (e.g., one child or one automobile). Intuitive scientists must also deal with considerable ambiguity in the relations between hypotheses and data. Typically, the magnitude of the measurement error is nearly the same as the magnitude of the effects under study. Hypotheses have ambiguous implications about the phenomena that should or should not be observed, and observed data have ambiguous implications as to how or whether a hypothesis should be revised.

The fundamental difference between the law-based prototype of science and the task of reasoning in daily life is the degree of uncertainty and complexity in the environment. By uncertainty, I mean that phenomena must be regarded as probabilistic, and not subject to complete explanation or prediction. By complexity, I mean that phenomena arise from the simultaneous influence of numerous and often inscrutable contributory factors. Although I focus on informal reasoning, many professional scientists also face high uncertainty and complexity, particularly in "soft" or "inexact" fields, like the social sciences, or in newer, less well-established domains such as high-temperature superconductors today.

Uncertainty and complexity have important implications for the kinds of hypotheses people form. In lawful domains, it may be a reasonable approximation to say that scientists search for the "true" explanation. In inexact domains, no hypothesis is expected to yield nearly perfect prediction or nearly complete explanation. Instead, the hypothesis may state that there is an association between two things, or that a certain factor should have a significant effect on a behavior of interest. The goal is not to discover the right rule or law. The goal may instead be the development of a theory that is "pretty good" according to domain-specific standards, or one that is significantly better than the previous hypothesis, or the goal may be merely to achieve predictive accuracy better than chance.

Uncertainty and complexity have parallel effects on the process of testing hypotheses. With probabilistic hypotheses, there

are no logically determined "critical" tests, nor any logical mandate as to how to modify the hypothesis in response to new data. Thus, the processes of hypothesis testing and revision become matters of accumulation and interpretation of evidence, and a process of zeroing in on a hypothesis that meets the hypothesis-tester's goal.

There is a fairly extensive body of research on the psychological processes of hypothesis testing, and this work has produced a number of interesting findings concerning the abilities and failings of human hypothesis testers (see Klayman & Ha, 1987; Nisbett & Ross, 1980; Wason & Johnson-Laird, 1972). In this paper, however, I focus on an area of research not usually associated with the study of scientific reasoning. This research has gone under a variety of names, but can be referred to generically as *cue learning*. Cue-learning tasks require the subject to make judgments based on one or more cues that provide only partial and imperfect information, and to use feedback to try to improve the accuracy of those judgments. Thus, cue learning captures the flavor of everyday reasoning better than many hypothesis testing tasks.

Cue learning

The development of cue learning in the 1950's was based on Egon Brunswik's (1956) principle of "probabilistic functionalism," the principle that judgments in natural environments must be derived from a combination of multiple, imperfect cues. Thus, the central goal of the paradigm is to study how people learn to relate cues to judgments in probabilistic environments. Cue-learning tasks have three basic elements: a criterion (something the subject must learn to predict or estimate), cues (information from which to make the estimate) and feedback (information about the accuracy of the estimates made). As in natural learning environments, the criterion is not fully predictable from the available cues. For example, the criterion value, Y, might be determined by the formula $Y = 1/3\ A + 2/3\ B + C + e$, where A, B, and C are cues, and e is a number from a random number generator.

Cue learning actually encompasses several different kinds of learning. First, there is the matter of "cue discovery" (Klayman, 1988), the process of finding valid cues to use in making predictions. Then, given a set of cues, there is the matter of how those cues should be combined. This includes determining whether effects are additive or multiplicative, and the relative importance weight to give each cue. Then, there is the task of determining the form of the different cue-criterion functions. For example, it may be that, other things being equal, the criterion has a positive linear relation to cue A, a negative linear relation to cue B, and a U-shaped relation to C.

Cue learning captures some of the complexity and uncertainty of real-life hypothesis testing. The behavior of interest is a function of a number of simultaneous factors, there is only a corelational association between any cue and the criterion, and all the available cues, taken together, cannot completely predict or explain the dependent measure. Furthermore, the feedback one gets is ambiguous in the sense that a discrepant finding may reveal an inaccuracy in the model you are using, or it may be attributable to random error; if a change is indicated, it is not clear just what the change should be. Thus, it should perhaps be no surprise that many cue-learning studies have found that it is difficult for people to learn from feedback in such situations (see Brehmer, 1980; Klayman, in press).

Hypotheses in cue learning

During the first couple of decades of cue-learning research, not much attention was paid to the matter of hypotheses. Cue learning was regarded as a process of learning to associate certain criterion values with certain values of each cue,

along with some process of averaging and interpolation. However, there is a growing body of evidence about the crucial role that hypotheses play in learning in complex and uncertain environments. In this regard, there is now an important bridge between cue-learning research and more mainstream research in learning and scientific reasoning.

Where do the hypotheses come from? Except when the learning task is presented abstractly (e.g., with cues identified only as *A*, *B*, and *C*), hypotheses will of course be derived from the learner's knowledge and theories about the causal structure of the environment. World knowledge may suggest specific functions (e.g., that the relation between effort and performance in a task is a positive one, with diminishing returns), or provide more general hints (e.g., to look at personality variables in this situation). At the most general level, one may fall back on general cues to causality such as temporal and spatial proximity (see Einhorn & Hogarth, 1986).

On the other hand, there is also evidence of a general default hierarchy of hypotheses that follows a sort of principle of intuitive parsimony. Given several cues from which to make judgments, subjects hypothesize mostly about how the cues ought to be combined, and they seem to pay little attention to the matter of cue-criterion functions (Brehmer, 1987). However, if they use the cues to make estimates in the meantime, they *must* make some assumptions about the underlying functions, at least de facto. Brehmer found that subjects' responses implied a default assumption of simple linear cue-criterion relations. This is also the most common initial hypothesis about the cue-criterion function in one-cue tasks (Brehmer, 1974). In a task involving cue discovery, in which the set of useful cues was not fully specified in advance (Klayman, 1988), subjects' hypotheses were almost exclusively in the form of "the more of this, the more of [or the less of] that." Subjects seldom expressed any hypotheses about how *much* of this or how much of that, and there were few hypotheses about interactions among cues. Responses implied a default assumption of linear cues that combined additively.

There is little evidence that intuitive parsimony is a conscious strategic principle. Rather, it can be viewed as an outcome of the feedback-encoding process. A simple way to encode feedback is to observe that a change in a cue tends to be associated with some direction of change in the criterion ("this one had more achievement motivation and did worse on the test"). This level of encoding permits cue discovery, since the learner could perceive the existence of an effect. Slightly more complex encoding might include some information about the magnitude of change ("...a lot more achievement motivation and did a little worse..."). This yields information about the average magnitude of the relationship, but nothing about its shape. The only way to recognize a nonlinear relation is to keep track of the relation between the *magnitude of changes* and the *absolute magnitude* of the cue. For example, the relation $Y = \log(X)$ implies that the larger X is, the smaller the change in Y with a given change in X. Similarly, perception of interactions would require encoding the relation between X_1 and Y as a function of X_2. Nonmonotonic functions and disordinal interactions may be particularly hard to learn, because even the *direction* of change will be observed to vary, and the learner may conclude that no consistent relationship exists.

Evidence from a number of cue-learning studies supports this ordering of task difficulty: People learn the identity of cues before they learn how best to combine them; they learn additive combinations more easily than others; and they learn linear relations more easily than nonlinear ones (see Klayman, in press). Part of the difficulty may be that people simply fail to consider hypotheses further down their hierarchy, but it is also the case that the more complex functions are simply more

difficult to perceive (Brehmer, 1980). The default hierarchy of hypotheses also interacts with world knowledge. On the one hand, some of the more complex functions, such as nonmonotonic cues and disordinal interactions, may be learnable if one has a prior hypothesis to guide the encoding of the data. On the other hand, people seem prone to encode their world knowledge in terms of simpler functions and combinations as well (Klayman, in press; Sniezek, 1986).

Testing hypotheses

In the law-discovery model of reasoning, investigators can reasonably expect to determine whether their hypotheses are right or wrong. (At least, following Popper (1959), you should be able to determine whether or not your hypothesis has been falsified yet). In cue learning, though, the goal is not to find out if the hypothesis is wrong (because it always is), but where it is wrong, and how it might be fixed. Hypothesis testing and revision is thus a process of gradual refinement, starting with general ideas about the types of things to consider, and moving to more complete and specific (and hopefully more accurate) hypotheses (see, e.g., Klayman, 1988; Klahr & Dunbar, 1988; Lakatos, 1978).

How then is feedback used to test hypotheses derived from world knowledge, previous feedback, and default rules? Research on hypothesis-testing behavior suggests that hypothesis testing under conditions of uncertainty and complexity is a very difficult task. A number of these difficulties have come under the rubric of "perseverance of beliefs" (Ross & Lepper, 1980) or "confirmation bias" (Fischhoff & Beyth-Marom, 1983; Klayman & Ha, 1987). For example, people tend to interpret ambiguous evidence in a way that favors their current hypothesis. In a probabilistic environment, feedback is always ambiguous is that it is never clear whether deviations from the expected are meaningful or "just" random. People may "immunize" their hypotheses, by accepting results that conform to their hypotheses, while attributing unexpected findings to random error more than is justified (Gorman, 1986). People also use a "positive test strategy" in which attention is focused on the ability of the current hypothesis to predict and explain observed events, with little consideration given to possible alternative hypotheses (Klayman & Ha, 1987, 1988).

The general implication of these aspects of hypothesis testing is that subjects will be slow to reject early hypotheses. This need not always be the case, however. In some situations, people seem very prone to changing hypotheses, and may hurt themselves by rejecting good ones. This will happen if learners have unrealistic expectations about how good a good hypothesis ought to be, i.e., if they underestimate the impact of hidden variables and random error. A number of studies have documented people's tendency to underestimate the role of chance (see Langer, 1975, for example), especially in the absence of world knowledge about the underlying mechanisms (Nisbett et al., 1983). The result can be a fickle hypothesis tester, who rejects and replaces hypotheses on the basis of insufficient negative evidence. This pattern has been observed in a variety of learning studies (Brehmer, 1980; Mynatt, Doherty & Tweney, 1978).

Hypotheses and learning from feedback

The real object of cue learning, and learning from experience in general, is not so much to test hypotheses, but to revise and improve them. From the above discussion, it might appear that when data meet hypothesis, the prospects for appropriate learning are poor. However, the use of feedback to choose and revise hypotheses can have different outcomes, depending on the relation between hypotheses and incoming data.

Not surprisingly, people make more accurate judgments when their hypotheses are

congruent with the data, for example, when the cue labeled "monthly debt" is negatively related to "credit rating" (Muchinsky and Dudycha, 1975). One straightforward explanation for this finding is that people do not *need* to learn from outcome feedback if they already have appropriate hypotheses. However, evidence also suggests that congruent hypotheses can facilitate subsequent learning. For example, Camerer (1981) found that subjects learned to use a multiplicative interaction present in outcome feedback when dimensions were labeled in a way that suggested the interaction. (MBA students perceived an interaction between price changes and trade volume in predicting a commodities market.) In contrast, subjects who were not given feedback did not manifest any such interaction in their estimates, nor did subjects given feedback with unlabeled cues.

Without a concrete hypothesis, subjects face the task of learning the associations between myriad cue values and a whole range of criterion values. Uncertainty and complexity make this abstraction process all the more difficult. An appropriate hypothesis can provide a useful way to organize feedback in encoding, aggregation, and recall. Data can be encoded as supporting or contradicting the hypothesis, and past experiences can be summarized in terms of a limited number of prior hypotheses, rather than a large number of individual feedback data. If a basic hypothesis about a cue seems valid, learners may then be able to use feedback to refine their ideas about the shape of the function and its relation to other cues.

But what if feedback in a learning situation *contradicts* the expectations you bring to it? One might expect misleading hypotheses to seriously interfere with learning, since good but counterintuitive hypotheses may never be considered, and poor but plausible hypotheses may persevere. Indeed, several studies have found that tasks that elicit inaccurate hypotheses are as hard to learn as purely abstract ones (Miller, 1971; Camerer, 1981) or harder (Adelman, 1981). On the other hand, some studies find that the information-processing benefits of a concrete hypothesis may even override misleading expectations. For example, Muchinsky and Dudycha (1975) and Sniezek (1986) report that subjects learned meaningfully labelled relations better than abstract ones even when the data seemed anomalous. In such cases, subjects ad libbed new hypotheses or reinterpreted the data, and then used the new interpretations to encode subsequent feedback. For example, some of Sniezek's subjects invented convoluted meteorological theories to interpret data suggesting that temperature *increased* as one got further north of the equator.

Conclusions

People are constantly forming, testing, and revising hypotheses about how the world works, what will happen next, or what the consequences of an action will be. This informal scientific reasoning differs in important ways from the prototype of science as the discovery of laws. Theories about everyday phenomena are of limited explanatory power and scope, and data are prone to considerable error and ambiguity. As research on cue learning illustrates, uncertainty and complexity in the environment affect the nature of hypotheses, the goals of hypothesis testing, and the processes by which data are encoded, aggregated, and interpreted.

Of course, uncertainty and complexity are encountered in the formal practice of science as well. However, the professional scientist is in a position to use tools such as controlled experimentation and statistical methods. These techniques certainly help with the problems of testing and revising hypotheses in a probabilistic environment. Even informal settings sometimes permit experimentation and quantification. Educated laypeople have a fair degree of intuition about some basic principles of experimentation, and they may learn and

reason more effectively when they are able to apply those principles (Klayman, 1988; Nisbett et al., 1983).

On the other hand, the availability of scientific methods does not eliminate the difficulties of using ambiguous data to test and revise hypotheses. The conduct of science is seldom as neat and clear as the law-discovery prototype, or as the resulting published articles make it sound (see, e.g., Mitroff, 1974). Informal thinking plays an important role in forming new theories and hypotheses, choosing and evaluating research methods, and interpreting findings. Thus, many of the phenomena of informal reasoning are likely to have relevance to the professional conduct of science as well, especially in less exact domains.

A thorough understanding of informal scientific reasoning is still a long way off. However, there is some convergence in recent work using a variety of different paradigms: Hypothesis testing is being viewed in a broader context, as one of the critical steps in an interactive process of discovery, testing, and revision of ideas (Holland, et al., 1986; Johnson-Laird, 1983; Klahr & Dunbar, 1988; Klayman & Ha, 1987; Lakatos, 1978; Langley et al., 1987). The course of this reasoning process is a function of world knowledge, prior theories, basic processing characteristics, and information from data. This view of hypothesis testing is much more complex than earlier models, but it is also more likely to be at least half right.

References

Adelman, L. (1981). The influence of formal, substantive, and contextual task properties on the relative effectiveness of different forms of feedback in multiple-cue probability learning tasks. *Organizational Behavior and Human Performance*, *27*, 423-442.

Brehmer, B. (1974). Hypotheses about relations between scaled variables in the learning of probabilistic inference tasks. *Organizational Behavior and Human Performance*, *11*, 1-27.

Brehmer, B. (1980). In one word: not from experience. *Acta Psychologica*, *45*, 223-241.

Brehmer, B. (1987). Note on subjects' hypotheses in multiple-cue probability learning. *Organizational Behavior and Human Decision Processes*, *40*, 323-329.

Brunswik, E. (1956). *Perception and the representative design of psychological experiments* (2nd Ed.). Berkeley: University of California Press.

Camerer, C. (1981). *The validity and utility of expert judgment*. Unpublished doctoral dissertation, University of Chicago, Graduate School of Business.

Einhorn, H.J. & Hogarth, R.M. (1986). Judging probable cause. *Psychological Bulletin*, *99*, 3-19.

Fischhoff, B. & Beyth-Marom, R. (1983). Hypothesis evaluation from a Bayesian perspective. *Psychological Review*, *90*, 239-260.

Gorman, M. E. (1986). How the possibility of error affects falsification on a task that models scientific problem-solving. *British Journal of Psychology*, *77*, 85-96.

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

Johnson-Laird, P. N. (1983). *Mental models*. Cambridge, MA: Harvard University Press.

Klahr, D. & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Psychology*, *12*, 1-48.

Klayman, J. (1988). Cue discovery in probabilistic environments: Uncertainty and experimentation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *14*, 317-330.

Klayman, J. (in press). On the how and why (not) of learning from outcomes. In B. Brehmer & C. R. B. Joyce (Eds.), *Human Judgment: The Social Judgment Theory Approach*. Amsterdam: North-Holland.

Klayman, J. & Ha, Y. (1987). Confirmation, disconfirmation, and information in hypothesis testing. *Psychological Review, 94*, 211-228.

Klayman, J. & Ha, Y. (1988). *Hypothesis testing in rule discovery: Strategy and structure* (Working Paper No.133). Chicago: University of Chicago, Graduate School of Business, Center for Decision Research.

Lakatos, I. (1978). *The methodology of scientific research programmes*. London: Cambridge University Press.

Langer, E. J. (1975). The illusion of control. *Journal of Personality and Social Psychology*, 32, 311-328.

Langley, P., Simon, H. A., Bradshaw, G. L., & Zytkow, J. M. (1987). *Scientific discovery: Computational explorations of the creative processes*. Cambridge, MA: MIT Press.

Miller, P.McC. (1971). Do labels mislead? A multiple cue study, within the framework of Brunswick's probabilistic functionalism. *Organizational Behavior and Human Performance, 6*, 480-500.

Mitroff, I. (1974). *The subjective side of science*. Amsterdam: Elsevier.

Muchinsky, P.M. & Dudycha, A.L. (1975). Human inference behavior in abstract and meaningful environments. *Organizational Behavior and Human Performance, 13*, 377-391.

Mynatt, C.R., Doherty, M.E. & Tweney, R.D. (1978). Consequences of confirmation and disconfirmation in a simulated research environment. *Quarterly Journal of Experimental Psychology, 30*, 395-406.

Nisbett, R. E., Krantz, D. H., Jepson, C., & Kunda, Z. (1983). The use of statistical heuristics in everyday inductive reasoning. *Psychological Review, 90*, 339-363.

Nisbett, R. E. & Ross, L. (1980). *Human inference: Strategies and shortcomings of social judgment*. Englewood Cliffs, NJ: Prentice-Hall.

Popper, K. R. (1959). *The logic of scientific discovery*. New York: Basic Books.

Ross, L. & Lepper, M.R. (1980). The perseverance of beliefs: Empirical and normative considerations. In R.A. Shweder (Ed.), *Fallible Judgment in Behavioral Research: New Directions for Methodology of Social and Behavioral Science* (Vol 4, pp. 17-36). San Francisco: Jossey-Bass.

Sniezek, J.A. (1986). The role of variable cue labels in cue probability learning tasks. *Organizational Behavior and Human Decision Processes, 38*, 141-161.

Tweney, R. D. (1985). Faraday's discovery of induction: A cognitive approach. In D. Gooding & F. James (Eds.), *Faraday rediscovered* (pp. 159-209). London: MacMillan.

Wason, P.C. & Johnson-Laird, P.N. (1972). *Psychology of Reasoning: Structure and Content*. London: Batsford.

Preparation of this article was supported by grant SES-8706101 from the Decision, Risk, and Management Science program of the National Science Foundation. Thanks to Jackie Gnepp and to my colleagues at the Center for Decision Research for their helpful comments.

A Theory of Scientific Problem Solving

Randolph Jones (RJONES@CIP.ICS.UCI.EDU)
Pat Langley (LANGLEY@CIP.ICS.UCI.EDU)
Irvine Computational Intelligence Project
Department of Information & Computer Science
University of California, Irvine CA 92717 USA

Introduction

We are interested in computational explanations of the nature of human problem solving. In the past, many artificial intelligence (AI) systems have implemented problem solving in a problem–space framework (Newell, 1980). In this paradigm, a problem consists of an initial state, a goal state, and a set of operators that can be used to transform the initial state into the goal state. These systems have been moderately successful in providing a formal analysis of problem solving, but they fail to exhibit many aspects of human cognition. We have developed a theory of problem solving that accounts for some of these phenomena. In addition, we have built EUREKA, a system that instantiates this theory, and we have tested the system in a variety of domains, including scientific reasoning tasks.

Characteristics of Human Problem Solving

We have developed our theory in an attempt to account for many of the characteristics of human problem solving. Thus, we will begin by discussing some of these characteristics.

Heuristic methods. Humans do not attack problems blindly. In particular, when a person encounters a new problem, he will not start applying all his knowledge in random patterns until he solves it. Rather, he uses heuristics, or educated guesses and rules of thumb, to guide his search for a solution. One type of systematic heuristic problem solving that has seen some success in AI is *means–ends analysis* (Ernst & Newell, 1969; Fikes & Nilsson, 1971). At each decision point, these systems attempt to apply an operator that reduces the differences between the current problem state and the goal. In this way, the search for a solution is directed down promising paths.

Non-systematic nature. However, humans do not solve problems in a very systematic manner. If a person finds himself stuck at some point, he can usually not remember all the steps he took in reaching that point. In many cases, he will simply start the problem again from the beginning, often duplicating previously failed paths in his new attempts. In contrast, most of the AI work on problem solving has employed memory-intensive methods, such as depth-first and best-first search. These techniques assume a large memory in which they can store all previous goals and states. Using this memory, they can 'backtrack' to any earlier point in the problem, as well as avoid duplicating past failures. Ohlsson (1987) has studied the non-systematic nature of human problem solving, but most current systems will attempt to explore their entire problem space systematically if they cannot find a solution.

Performance improvement and Einstellung. Humans learn while they solve problems. One aspect of this learning involves improved performance. This can be seen when a human

transfers knowledge and methods from a previous problem to solve a new problem more easily. In general, we expect humans to get better as they solve a set of similar problems (Ohlsson, 1987), but there are also instances when learning causes negative transfer. One example of this is *Einstellung*, or the 'set' effect. In this case, a person has become so used to solving problems in a certain way that he ignores a much simpler solution (Luchins, 1942; Neves & Anderson, 1981).

Response to external stimuli; Insight. Finally, humans are influenced by their environment. External cues can often aid a person in solving a problem (Dreistadt, 1969). Cues can also cause people to experience flashes of *insight.* Hadamard (1949) has detailed four stages in episodes of scientific insight. The first stage is *preparation*, in which a scientist works on a problem for some time with out success. When he gets frustrated and ceases work on the problem, he enters the *incubation* stage. Some time later (from a few seconds to a few years), *illumination* or insight occurs, during which a potential solution suddenly pops into the scientist's head. Finally, during *verification*, he works out the details of his solution. We have argued elsewhere (Langley & Jones, 1988) that illumination occurs when an environmental cue causes the scientist to suddenly retrieve an operator which will aid in solving his problem.

The EUREKA System

EUREKA is a running LISP program designed to model some of the processes of scientific discovery and problem solving. It consists of a memory component, a problem solver and a simple learning mechanism. The memory component can be described at various levels of abstraction, so we will begin with a low-level description and then discuss the higher level data representation as it applies to problem solving.

Memory representation and retrieval

EUREKA includes a long-term memory, represented as a semantic network consisting of nodes (concepts) connected by a small set of labeled links (relations). Each link has an associated *trace strength*, which represents the strength of the connection between the two attached nodes. For example, a link connecting 'bird' to 'wings' would probably have a larger trace strength than a link connecting 'bird' to 'legs'. EUREKA does not embody the notion of a specific short-term or working memory. However, each node also has associated with it a *level of activation*, which exhibits how much attention the concept receives during problem solving. Using this representation, retrieval of concepts is implemented as a form of *spreading activation* (Quillian, 1968; Anderson, 1976, 1983).[1] When a node is activated, it 'spreads' its activation to nearby nodes in the semantic network. As activation spreads from a node, this activation is divided up between all the connected nodes in proportion to the trace strengths of the links involved.

EUREKA's memory is further organized into conceptual units that are used by the problem-solving component. Each of these units consists of a collection of nodes in the semantic network. These collections include operators, problem-space states, and deriva-

[1] The type of spreading activation we use is a bit different from that introduced by Quillian, in which activation was used to find pathways between two concepts. Our approach is more similar to that used by Anderson (1976) and by Holland, Holyoak, Nisbett, & Thagard (1986) in their work on analogy.

tional trace structures. Operators and problem-space graphs are well-documented concepts in problem solving (Newell, 1980), but the third notion bears a more in-depth explanation. We have borrowed the idea of derivational traces from Carbonell (1986). These are records of problem-solving episodes in which information is stored about the system's goals and its reasons for making certain choices. Derivational traces can be used to remember the details of previous attempts to solve problems and to aid in solving similar problems.

In EUREKA, a derivational trace is represented as a tree. Each node in the tree represents a *goal*, and its children represent the subgoals that must be satisfied in order to achieve that goal. The system distinguishes between two types of goals: TRANSFORM a problem-space state into another state, or APPLY an operator to a problem-space state (Ernst & Newell, 1969). If a node in the derivational trace has no children, it means that either no subgoals were required to satisfy the goal (success) or no subgoals could be found which would help satisfy the goal (failure). During problem solving, activation is spread throughout the derivational trace structure and the rest of memory to aid in choosing operators.

Problem-solving component

EUREKA's basic problem-solving method is means-ends analysis, similar to that used in GPS (Ernst & Newell, 1969) and STRIPS (Fikes & Nilsson, 1971). A TRANSFORM goal can be satisfied by first applying an operator (i.e., setting up an APPLY goal) and then recursively TRANSFORMing the result. An APPLY goal can be satisfied by TRANSFORMing the current problem-space state to match the preconditions of the operator to be applied, and then APPLYing the operator to the resulting state. However, there are a few important differences.

Given a TRANSFORM goal, STRIPS would exhaustively search its set of operators and choose the 'best' one (using means-ends analysis) to APPLY. In contrast, EUREKA retrieves a small set of operators by spreading activation throughout its memory from nodes representing the current problem-space state and goal. The system then passes these operators on to a STRIPS-like matcher to decide which ones might be useful. The remaining operators are weighted according to how easily they were retrieved and how useful they have been in the past. Finally, the system selects a single choice at random based on these values. If no useful operators are found for a given goal, the problem solver fails.

Another important difference from STRIPS-like problem solvers is that EUREKA does not have the ability to backtrack. Instead, when the system fails to solve a problem, it starts over from the initial TRANSFORM goal for that problem. EUREKA attempts to solve the problem repeatedly, but it may duplicate previous problem-solving paths in the process. The model continues working until the problem is solved or until it becomes 'frustrated' and quits. Frustration occurs when the initial goal has a very high failure rate. Table 1 summarizes the the system's problem-solving component.

Learning and memory maintenance

In addition to standard problem-solving actions, EUREKA maintains a large derivational trace structure in its long-term memory. Whenever the system encounters a new situation during problem solving (e.g. a new problem-space state or a new derivation-trace node), it adds this situation to the derivational trace. In this fashion, a record of all previous problem

Table 1. EUREKA's basic problem-solving algorithm.

```
TRANSFORM(State1,State2)
If State1 satisfies State2
     Then Return(State1)
     Else Let Remindings be a set of instantiated operators retrieved
                through spreading activation;
          Let OpA be an operator selected at random from any
                Remindings that reduce differences between State1 and State2;
          If OpA is empty
               Then Return(Fail)
               Else Let State3 be APPLY(OpA,State1);
                    If State3 is Fail
                         Then Return(Fail)
                         Else Return(TRANSFORM(State3,State2))

APPLY(OpA,State1)
If OpA can be applied to State1
     Then Return(EXECUTE(OpA,State1))
     Else Let State2 be TRANSFORM(State1,Preconditions(OpA));
          If State2 is Fail
               Then Return(Fail)
               Else Return(APPLY(OpA,State2))
```

solving behavior is stored. If a new situation has already been stored in memory, the trace strengths of the links involved in that situation are increased slightly. In addition, special actions are taken upon the success or failure of a goal. Counts are kept to record how often each goal has succeeded or failed, and when a goal succeeds the trace strengths of the nodes involved in the goal are increased. These counts are used to estimate the probability that the goal will succeed or fail in the future.

Evaluation of the EUREKA Model

We have discussed a number of phenomena that our theory should handle. In order to test the theory, we have implemented it in the EUREKA system. Further, we have designed a number of problems in various domains to test the system along these lines. In this section we describe some experiments we have run with respect to the human problem-solving characteristics we described earlier.

Non-systematic, heuristic methods. EUREKA uses a variant of means-ends analysis that does not have the ability to backtrack. In addition, it uses spreading activation and counts of previous failures and successes to aid in conflict resolution. Since trace strengths are updated when familiar situations are encountered, the system can get stuck repeating old, unsuccessful behavior. However, the system tries to avoid previously failed states, so it can break out of this behavior. This encourages the system to explore a wide area in the problem space. We tested our system on a number of small 'blocks world' and 'chemical structure'[2] problems. EUREKA solved these problems after a small amount of exploration. When given more complicated problems (with no previous problem-solving memory), the system could not overcome the large problem space. It explored a large section of the space, but could

[2] These problems are based on Kekulé's problem of determining the structure of various molecules including benzene (Farber, 1966).

Table 2. 'Blocks-world' problems with two blocks.

Problem Goal	Length of Optimal Solution	Number of Attempts		Percentage of Space Searched	
		Without learning	With learning	Without learning	With learning
A over Table	3	—	1	—	5.0
B over Table	3	—	1	—	5.0
A over B	5	—	1	—	8.3
A on B	7	10	1	65.0	15.0

Table 3. 'Blocks-world' problems with three blocks.

Problem Goal	Length of Optimal Solution	Number of Attempts		Percentage of Space Searched	
		Without learning	With learning	Without learning	With learning
A over Table	3	—	3	—	6.2
B over Table	3	—	1	—	0.6
C over Table	3	—	1	—	0.6
A over B	5	—	2	—	4.0
B over C	5	—	1	—	1.0
B on C	7	—	1	—	1.4
A over B on C	11	—	1	—	2.2
A on B on C	13	10	1	30.0	2.6

not find the solution paths.

Performance improvement. EUREKA does have the ability to improve its performance based on previous experiences. We gave the system the same sets of 'blocks-world' and 'chemical-structure' problems. However, this time we ordered the problems from simplest to hardest and did not erase the system's memory after each problem. In this case, the system was able to solve all problems presented to it within three attempts. Tables 2 and 3 provide data from runs in the 'blocks-world' domain. Within each table, the initial states are the same. The optimal solution length is the size of the smallest derivation trace required to solve the problem.

Einstellung. Our theory explains Einstellung in terms of the trace strengths on links in the semantic network and the success counts kept for each state. Recall that trace strengths are increased whenever a goal is satisfied. This causes the system to retrieve the successful operators in similar situations. Combined with the record of success counts, this encourages the system to duplicate past successful behavior in new, similar situations. To test this effect, we gave EUREKA a series of 'water jug' problems (Luchins, 1942). The first few problems

required similar solution paths, and EUREKA exhibited improvement in duplicating this path with each new problem. The last problem could be solved using the same solution, or by using a unique solution that required only one operator application. The system chose to duplicate the solution path it had become familiar with from the previous problems, as humans often do in such situations.

Response to external stimuli. As shown in the earlier, there are times when EUREKA cannot solve a given problem, even if it has all the appropriate operators stored in memory. This can arise because the problem space is too large or because the appropriate operators are never retrieved from memory. However, the system can solve these problems in the presence of the appropriate external cues. We tested the system with the previous difficult problems, and with a problem simulating Archimedes' discovery of the principle of displacement[3] (Dreistadt, 1968). During each of these problems we activated useful concepts in long-term memory. The activation from these concepts caused the appropriate operators to be chosen to solve the problems. The system was thus able to solve problems it could not normally solve without cues from the environment. We feel this provides an initial account of the illumination stage of scientific insight. These experiments exhibit how a person might be unable to solve a problem and then suddenly discover a solution.

Discussion and Future Work

We have presented a number of characteristics of human problem solving and a theory that accounts for them. These include heuristic and non-systematic problem solving, performance improvement, Einstellung, and stimulus-driven problem solving. Our theory explains these characteristics in terms of memory retrieval. We have implemented this theory in a computer simulation called EUREKA, and have tested its behavior in a number of situations. The results of these studies indicate that the model accounts for the characteristics we have discussed. We feel these results lend support to our explanation of human problem solving in terms of memory retrieval.

We plan to extend our theory and the EUREKA system to account for analogical problem solving and episodes of insight. We have shown that spreading activation has certain properties that allow transfer in problem-solving behavior. We believe that the same properties can be exploited to suggest and elaborate analogical solutions based on previous problems. We also believe that we will be able to account for insightful problem-solving experiences like those of Kekulé, Archimedes, and Darwin (Barlow, 1959). Elsewhere (Langley & Jones, 1988) we have described insight as a combination of effects from analogy and memory-limited problem solving. We plan to build these ideas into the EUREKA system, and we are hopeful that the results of future experimentation will further support our theory.

[3] In this problem, the goal is to prove that a crown is made of pure gold. The solution method involves determining the volume of the crown and comparing that to the volume of a gold brick of the same weight. However, the crown's volume cannot be measured by melting it down, because that would destroy the crown. Archimedes' key operator, which resulted in the formulation of the principle of displacement, was to measure the volume of the crown by immersing it in water and measuring the amount of water displaced.

References

Anderson, J. R. (1976). *Language, memory, and thought.* Hillsdale, NJ: Lawrence Erlbaum.

Anderson, J. R. (1983). *The architecture of cognition.* Cambridge, MA: Harvard University Press.

Barlow, N. (1959). *The autobiography of Charles Darwin.* New York: Harcourt Brace.

Carbonell, J. G. (1986). Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach, volume 2* (pp. 371–392). Los Altos, CA: Morgan Kaufmann.

Dreistadt, R. (1968). An analysis of the use of analogies and metaphors in science. *The Journal of Psychology, 68*, 97–116.

Dreistadt, R. (1969). The use of analogies and incubation in obtaining insights in creative problem solving. *The Journal of Psychology, 71*, 159–175.

Ernst, G., & Newell, A. (1969). *GPS: A case study in generality and problem solving.* New York: Academic Press.

Farber, E. (1966). Dreams and visions in a century of chemistry. In R. F. Gould (Ed.), *Kekulé centennial* (pp. 129–139). Washington, DC: American Chemical Society.

Fikes, R. E., & Nilsson, N. J. (1971). STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence, 2*, 189–208.

Hadamard, J. (1949). *The psychology of invention in the mathematical field.* Princeton, NJ: Princeton University Press.

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

Langley, P., & Jones, R. (1988). A computational model of scientific insight. In R. Sternberg (Ed.), *The nature of creativity* (pp. 177–201). Cambridge, England: Cambridge University Press.

Luchins, A. S. (1942). Mechanization in problem solving: The effect of Einstellung. *Psychological Monographs, 54*(248).

Neves, D. M. & Anderson, J. R. (1981). Knowledge compilation: Mechanisms for the automatization of cognitive skills. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 57–84). Hillsdale, NJ: Lawrence Erlbaum.

Newell, A. (1980). Reasoning, problem solving, and decision processes: The problem space hypothesis. In R. Nickerson (Ed.), *Attention and performance VIII.* Hillsdale, NJ: Lawrence Erlbaum.

Ohlsson, S. (1987). Transfer of training in procedural learning: A matter of conjectures and refutations? In L. Bolc (Ed.), *Computational models of learning* (pp. 55–88). Berlin: Springer-Verlag.

Quillian, M. R. (1968). Semantic memory. In M. L. Minsky (Ed.), *Semantic information processing.* Cambridge, MA: MIT Press.

EMPIRICAL ANALYSES AND CONNECTIONIST MODELING OF REAL-TIME HUMAN IMAGE UNDERSTANDING

IRVING BIEDERMAN, THOMAS W. BLICKLE, GINNY JU, H. JOHN HILTON, AND JOHN E. HUMMEL

UNIVERSITY OF MINNESOTA

In less than 100 msec, humans can accurately interpret images of objects and scenes that have never been experienced previously, or that are extensively degraded, or are viewed from a novel orientation. Recent research and theory (Biederman, 1987a, b, c) suggest that this achievement may be based on a process that decomposes complex visual entities into simple components, typically at regions of matched concavities. Such concavities are almost always produced when shapes are arbitrarily joined (Hoffman & Richards, 1985). The resultant components activate the closest fitting member of a particular set of convex or singly-concave edge-based volumetric primitives, called geons, that are invariant under changes in viewpoint and visual noise and allow objects so represented to possess the same invariance. The geons require only categorical classification of edge characteristics (e.g., straight vs curved; parallel vs nonparallel; vertex type) rather than precise metric specification (e.g., degree of curvature or length of an edge). The latter type of judgments cannot be made with sufficient speed or accuracy by humans to be the controlling processes for real time human object recognition.

The capacity to represent the 10^6 objects that people can rapidly classify derives from an allowance of several viewpoint invariant relations (e.g, TOP-OF, CENTER-CONNECTED) defined for joined pairs of geons such that the same subset of geons represent different objects if they are in different relations to each other. A description of the input consisting of geons + relations is matched against a similar type of description in memory. For example, one kind of lamp can be described as a cylinder CENTERED UNDER THE LARGER END of a cone. Matching is graded in that the activation of a representation will be slower (and of lower maximum value) when image descriptions differ in geons or relations. Geons thus play a role highly analogous to the role played by phonemes in speech perception.

A *Principle of Geon Recovery*, derived from the theory, can account for the major phenomena of object recognition: If an arrangement of two or three geons can be recovered from the image, objects can be quickly recognized even when they are occluded, rotated in depth, novel, extensively degraded, or lacking customary detail, color, and texture.

Empirical Studies of Human Image Understanding

An extensive series of experiments on the perception of briefly presented pictures by human observers has provided empirical support for the theory. In these experiments the subject names or verifies briefly presented (100 msec.) object pictures. Reaction times and errors are the primary dependent variables. Some key results:

1. Simple line drawings showing only the edges of the major geons are identified as rapidly as full color, textured images (Biederman & Ju, 1988). This documents the sufficiency of edge-based descriptions in accounting for the initial activation of a representation of an object.

2. When only two or three geons of a a complex object (such as an airplane or elephant) are visible, recognition can be fast and accurate (though, predictably, not as fast as with the complete image). This supports the derivation of the sufficiency of three geons.

3. Complex objects requiring six or more geons to appear complete are not recognized any more slowly than simple objects (such as a flashlight or cup). This is consistent with a model positing parallel activation of the geons in favor of a serial contour tracing process, such as eyemovements or the kinds of serial routines posited by Ullman (1984).

4. If contour is deleted so that an object's geons cannot be recovered from the image (by deleting cusps for parsing and altering vertices) the object is rendered unrecognizable. If the same or greater amount of contour is deleted but in such a manner that the geons can be recovered through smooth continuation, objects remain identifiable. This result establishes the necessity of the contours posited by RBC.

5. A surprising finding in the previous experiment was the large disruptive effect on error rates and reaction times of interrupting (deleting) contour, such as would be produced when viewing an object behind light foliage, even when the contour could be restored by routines for smooth continuation. This suggests that the routines for contour restoration are not particularly rapid.

6. In the studies described in the previous paragraph, the contour that was removed was removed from every geon in the object. Identification performance is also slowed when objects are missing geons (parts) with the rest of the object intact, such as would occur if the object was partially occluded by a solid surface. According to the theory, the effect of missing or occluded geons is on the matching stage, rather than on the initial determination of the geons.

7. Rotation of the object in the plane slows recognition to a much greater extent than rotation in depth (in contrast to most robot vision models). According to the theory, rotation in the plane affects the TOP-OF relation but the geon descriptions themselves are largely unaffected by rotation in depth.

8. *Complementary* images of objects, in which alternative vertices and edges have been deleted, so that the composite will reveal the original intact image, as illustrated in the upper portion of figure 1, are treated equivalently. This suggests that the memorial representation can be described in terms of geons rather than the precise image features that elicited the geons.

A Connectionist Model of RBC

Hummel, Biederman, Gerhardstein and Hilton (1988) are implementing a connectionist model of RBC as shown in Figure 2. The model is a three layered network which takes as its input an activation vector representing the vertices and cusps in the image of an object as shown in Figure 3. The model gives as output an activation vector representing a geon-based description of the object from which the image was derived. Retinotopic mapping is preserved in all three of the model's layers. Given the spatially specified vertex and edge descriptions in the input vector, a major goal of this effort is to determine: a) if parsing of an image of an object into its constituent geons can be achieved, and b) if the spatial relations among the geons and the global properties of the geons themselves (i.e., parallelism, symmetry), can be derived.

The model's first layer is organized into 182 hexagonally arranged edge and vertex detectors at three spatial scales (20' [N=138], 40' [N=37], and 80' [N=7]). The receptive fields of adjacent detectors overlap and, together, span the central 4° of the visual field. Each of the detectors has 26 nodes for expressing the various vertices (Y, Arrow, Tangent Y, L, T), the edge types comprising them (straight, curved, or cusp) at one of eight orientations.

The next (or hidden) layer is organized into 119 100-node clusters, each of which is termed a geon field. Each geon field receives input from seven contiguous feature detectors at a given scale and passes input, 1:1, to a corresponding geon field in the upper layer. Each detector is mapped to its seven contiguous geon fields at the appropriate scale. The set of seven detectors

mapped to their middle and upper layer geon fields is termed a *column*. The representation in the lower and upper layers have been designed a priori to express general assumptions about edge coding and RBC's object representations, respectively. As the model is being trained to recognize objects through back propagation, the representation in the middle layer will emerge as a function of the constraints of the mapping between the lower and upper layers.

The columns all have identical connection matrices and no connections exist between columns. In this manner, the response of the system will be identical independent of where an object happens to fall in the visual field. A significant economy in the number of connections results from this columnar organization. In total the model contains 17,932 nodes (4,732 in the first layer [26 nodes per feature detector X 182 detectors], 12,000 in the middle layer [100 per geon field for 120 geon fields], and 1,200 in the top layer [10 per geon field for 120 geon fields]). With the system fully interconnected this would result in 71,184,000 connections. But the columnar restriction results in only 2,304,000 connections, a savings of 96.76% in the number of connections.

The upper layer codes distributed representations of geons by locally representing geon attributes, such as whether the cross section is straight or curved. In this layer relations among the geons are represented implicitly in terms of the spatial relations among the patterns of activation representing those geons. The distributed coding of geons at this layer produces an object-centered representation.

REFERENCES

Biederman, I. (1987a). Recognition-by-Components: A Theory of Human Image Understanding. Psychological Review, 94, 115-147.

Biederman, I. (1987b). Matching Image Edges to Object Memory. In Proceedings of the First International Conference on Computer Vision, IEEE Computer Society, 384-392. London, England, June, 1987.

Hoffman, D. D., & Richards, W. (1985) Parts of recognition. Cognition, 18, 65-96.

Ullman, S. (1984). Visual routines. Cognition, 18, 97-159.

Figure 1. Complementary images of a single object. When viewed separately, these images are treated equivalently.

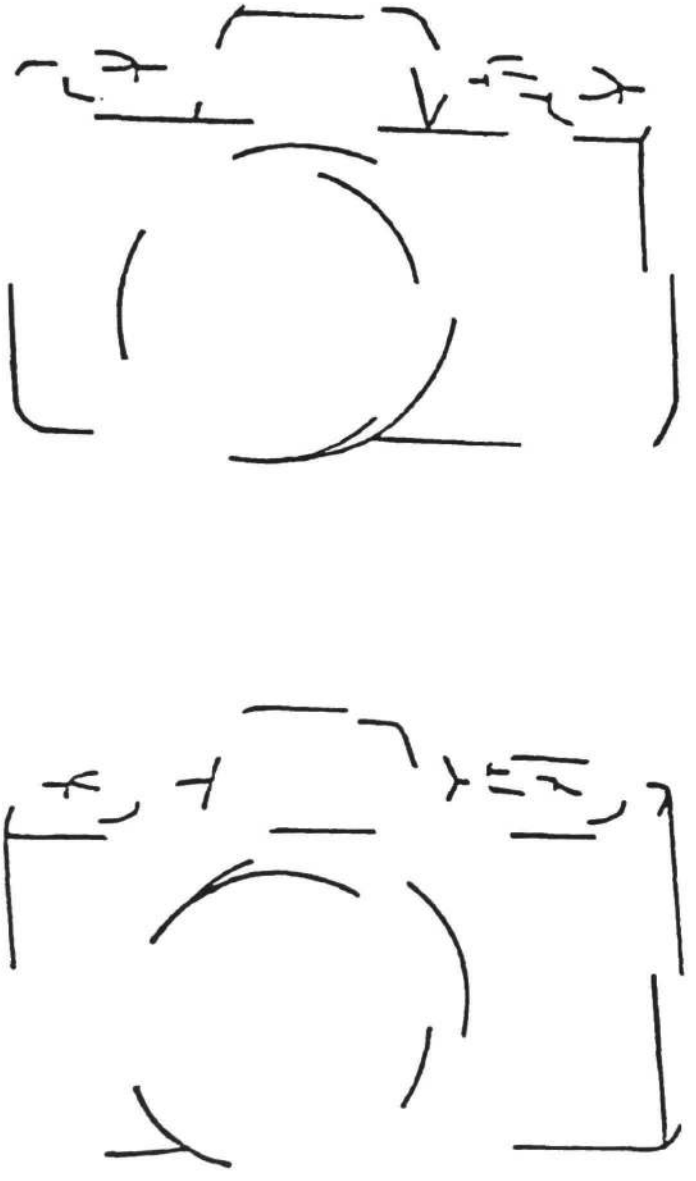


Figure 2. The connectionist model of object recognition.

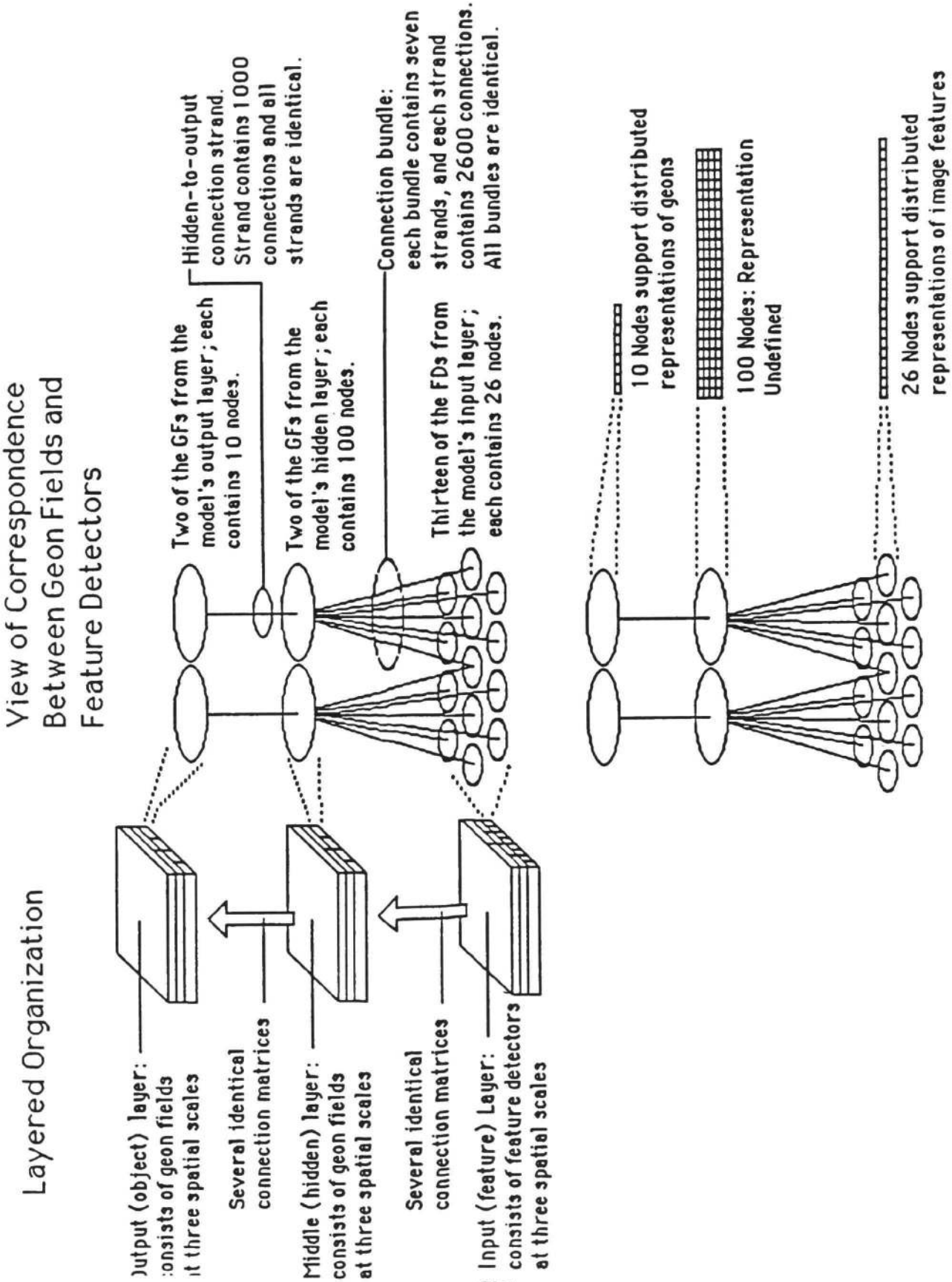
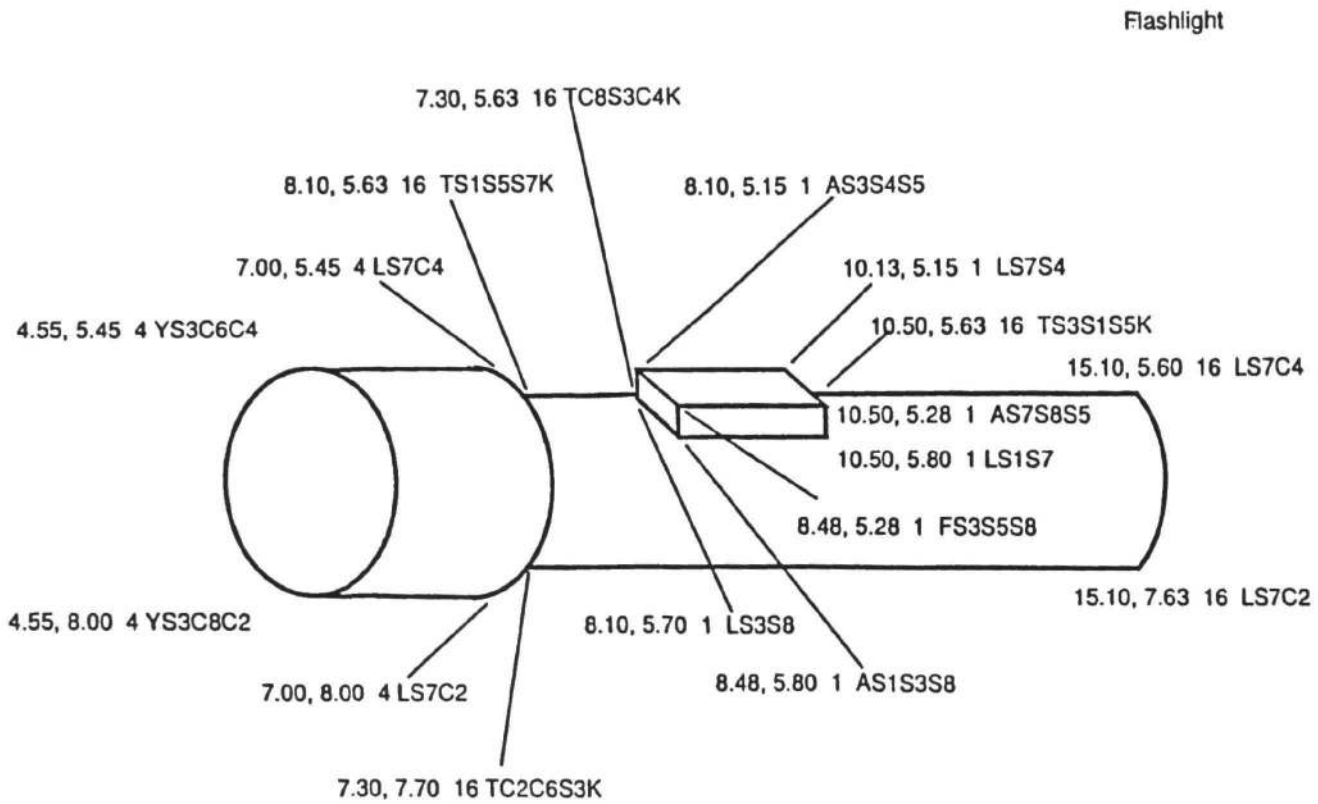


Figure 3. An example of the coding of a single object, a flashlight. The first two values by each vertex give the X and Y coordinates. The next value is the scale at which that vertex would be detected, the first letter provides the vertex type (A = Arrrow, F = Fork, L = L, T = T. The following letters and numbers specify the edge type (S = straight, C = Curved) and orientation of that edge (eight values). Cusps are designated with a K.



Learning to represent and understand locative prepositional phrases

Cynthia Cosic
Paul Munro

University of Pittsburgh
Department of Information Science

1. Introduction

One of the principal problems in natural language processing (NLP) is the role of context in understanding the sense of a particular word in a particular situation. Therefore the connectionist approach, which has fared quite well in accounting for context sensitivity, has become an attractive approach to NLP (eg. Cottrell & Small, 1983). This report presents a network that learns the spatial relationships that can be communicated through the use of a locative preposition. The goals of this network were to both "decode" and encode" prepositional phrases. (Herskovits, 1986) The decoding task involves understanding the spatial relationship between two nouns (N1 and N2) described by a locative preposition (LP); consider, for example, the different uses of the LP *in* accessed by the phrases *water in glass* and *crack in glass*; in the first case, N2 is acting as a container for N1 whereas in the second N1 is within the substance of N2. The encoding task is the converse problem: finding the appropriate preposition to express a given spatial relationship between two particular objects.

In addition to discovering whether the network could learn to encode and decode prepositional phrases, another area of interest was studying the context-free meanings of the prepositions themselves. The ability of connectionist networks to complete partial patterns (e.g. McClelland, Rummelhart, and Hinton, 1986) makes it possible to study the response of the network to a preposition alone, in the absence of the nouns. In this case, the question of whether these meanings were the same as the ideal meanings defined by Herskovits (1986) was investigated. Herskovits's (1986) theory of the meanings of locative prepositional phrases is built around the notion of an "ideal meaning", which is like a prototype. (Herskovits, 1986, p. 39) It is constructed out of the "perceptually salient" characteristics of a set of objects and their relationship (Herskovits, 1986, p. 54). Her ideal meanings of the prepositions *in*, *on*, and *at* are:

> The ideal meaning of *in*:
> The inclusion of a geometric construct in a one-, two-, or three-dimensional geometric construct. (Herskovits, 1986, p. 48)

> The ideal meaning of *on*:
> For a geometric construct X to be contiguous with a line or surface Y; if Y is the surface of an object Oy and X is the space occupied by another object Ox, for Oy to support Ox. (Herskovits, 1986, p. 49)

> The ideal meaning of *at*:
> For one point to coincide with another. (Herskovits, 1986, p. 50)

Finally, the way the network grouped the nouns used as arguments to the preposition was studied. The network was designed to build its own internal representation for each noun. Once again, Herskovits's (1986) theory of locative prepositional phrases provides clues as to now the nouns might be

divided by the system. While Herskovits feels that the ideal meaning is at the core of each use of a preposition, she also states that in actual use the meaning is often changed or shifted in some manner. (Herskovits, 1986, p. 39) A particular use of a prepositional phrase, then, often has a meaning similar to (but not exactly the same as) the ideal meaning of the preposition. The structure she uses to hold the relationship between an ideal meaning of a preposition and the meaning embodied in a particular instance of using a preposition is called a use type. (Herskovits, 1986, p. 87) We did not represent all the use types in our simulation; in particular, our current architecture cannot support those which require more than two arguments to the preposition, and we have chosen to exclude abstract relationships such as membership (e.g. student in school) in our initial studies. The use types that were represented in the stimulus set suggested possible ways that the network might categorize the nouns. These divisions are shown in Table 1.

Institution	
Gap	
Part/Whole	
Geometric Entity	Point Line Plane
Spatial Entity	Geographic Location Landmark Generic Location
Physical Object	Offers support Is a Media Has an Outline One of a group Container Usable Artifact Person

Table 1. Possible divisions of nouns, corresponding to distinctions required by Herskovits.

2. Network architecture

The network currently under study is feed-forward, with an input layer partitioned into four banks, three intermediate banks, and an output layer partitioned into two banks, organized as in Figure 1.

The network is a variation of the "encoder architecture" introduced by Ackley, Hinton, and Sejnowski (1986), which is trained to replicate patterns of input at the output layer. A pattern of the form N1-LP-N2 or N1-SP-N2 is presented to the network by activating from three to five units: one unit in bank A corresponding to N1, a noun; one unit in bank D corresponding to N2, a second noun; and either one unit in bank C, corresponding to the LP, the locative prepostion, or from one to

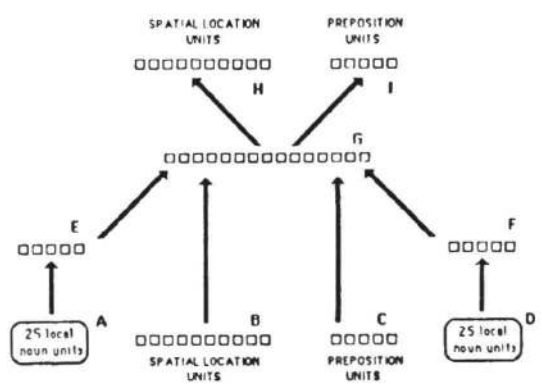


Figure 1. Network architecture for the preposition task. Inputs are presented at the lowest layer, across either banks A, B, and D or A, C, and D. The bold arrows indicate the flow of information from one bank to another. For example, every unit in bank F contributes to the activation of every unit in bank G. After each pattern presentation, the outputs at H and I are compared with the corresponding SP and LP components of the input pattern (banks B and C) to generate an error signal for the learning procedure.

three units in bank B, which correspond to the SP portion of the input, the units that describe the spatial relationship between the two nouns. Table 2 gives a list of all of the units that have been used. Because the network receives incomplete stimuli, and the output level is trained to supply both the LP and SP units, the network can be said to perform a pattern completion task.

Banks E and F receive information exclusively from banks A and D respectively. Therefore, the weight matrix from A to E (let this matrix be denoted [A,E]) must adapt such that representations of N1 are distributed in a manner that facilitates solution of the task. Similarly, F must come to give appropriate representations of N2. Careful study of these representations will be a major component of the task analysis. It should be noted that matrix [A,E] is constrained to be equal to matrix [D,F]; that is, a given noun comes to have the same representation in bank E as it has in bank F, so that important properties of nouns when they occur in the first position may carry over to situations where they occur in the second.

3. Simulation

3125 (25 x 5 x 25) pattern combinations can be formed with the twenty-five nouns and five prepositions; of these, 99 were chosen to constitute a "training corpus". During each training cycle, two input-output pairs were presented to the network. The first half-cycle consisted of presenting N1 on bank A, LP on bank C, and N2 on bank D; an error signal was generated by comparing the resulting response at H and I with the desired response, consisting of SP and LP respectively. This error signal was applied using the back-propagation procedure of Rumelhart, Hinton, and Williams (1986) to adaptively modify the weight values throughout the network. The second half-cycle used the same stimulus pattern except the input consisted of N1 on bank A, SP on bank B, and N2 on bank D. Each unit in the network responds as a semi-linear function of its input values; that is, the inputs were linearly summed to give a net activation value x, from which the response r was calculated using the logistic function:

$$r = \frac{1-e^{x}}{1+e^{x}}$$

Noun Units (25) Banks A and D	clouds	lake	campsite	table	book
	sky	river	school	glass	flowers
	plane	road	house	bowl	grass
	boat	city	floor	crack	man
	water	island	room	chip	fish
Semantic Units (10) Bank B	N1 over N2		N1 within border of N2		
	N2 over N1		N1 touching N2		
	N1 at edge of N2		N1 near N2		
	N1 embedded in N2		N1 far from N2		
	N2 contains N1		N2 supports N1		
Preposition Units (5) Bank C	in	at	on	under	above

Table 2. Unit labels for coding input patterns across banks A, B, C, and D

4. Results

The network required from 1,500 to 2,500 passes through the corpus to learn the prepositional phrases with an average error of less than 2%. Various tests were performed on the trained network in order to determine the ideal meaning of each preposition and the network's classification of the various nouns. Specifically, the question of whether the network would follow Herskovits's (1986) theory for the preposition's meanings and the types of nouns that can be arguments to the preposition was studied. In addition, the role of context in determining the meaning of a prepositional phrase, and the responses of the network to novel stimuli were investigated. The following paragraphs will deal with each of these findings separately.

After training, to test whether the network developed ideal meanings consistent with Herskovits (See Introduction), each LP (Bank C) was presented to the network alone, and the SP output (Bank H) was observed. The output representations are shown in Figure 2; filled rectangles indicate positive activity values, and open rectangles indicate negative activities.

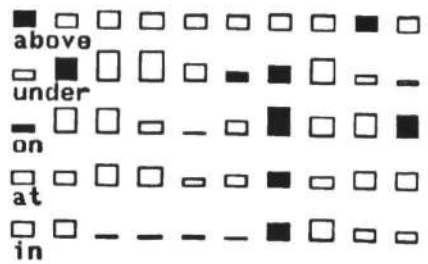


Figure 2. Bank I (spatial) output generated from prepositions alone. After training for from 150,000 to 250,000 pattern presentations (until a very low error is achieved by the network) the system is probed by observing the output of the H (SP) bank to each LP (bank C) input. The positively valued outputs (dark rectangles) correspond to: *in* = embedded in, and touching; *on* = 1 over 2, touching, and supported by; *at* = touching; *under* = 2 over 1, embedded, and touching; and *above* = 1 over 2 and far.

Note that during training, only the preposition *at*, always co-occurs with the same SP representation. While the meanings of *on*, *at*, and *above* do match those that would be expected, the meaning of *in* is not inclusion, as would be predicted. This is probably due to the high number of phrases that have to do with a physical object embedded in some medium, e.g. *the clouds in the sky*. The same is true for the preposition *under*, where the majority of the training examples were phrases such as, *the fish under the water*.

The distributed representations of the nouns were also studied. Because the weight matrices [A,E] and [D,F] were constrained to have the same values, the representation of a noun is the same in banks E and F; these are shown in Figure 3a. We were not able to determine a clear meaning for each unit in the distributed noun representations. However, the similarity structure of this representation was revealed through cluster analysis; the organization imposed upon the hidden units by this network via the learning procedure in order to solve the task (Figure 3b), correlates nicely with the one predicted from Herskovitz's theory (See Table 1).

The output of the network is context sensitive; many of the prepositional phrases used by the network do not have an output representation that corresponds to the preposition's ideal meaning. The output representation of a prepositional phrase is determined not only by the preposition, but also by the particular nouns used in the phrase. For example, the phrase *boat in water* is generally associated with the SP unit *N1 on top of N2* and *N1 supported by N2*, whereas *fish in water* fits much more nicely with *N1 embedded in N2*. To test context sensitivity, all the patterns of one noun and one preposition were presented to the network. At times, these representations are different from not only the preposition's ideal meaning, but also from the final meaning of a full prepositional phrase that incorporates that noun and that preposition. An example of this is shown in Figure 4.

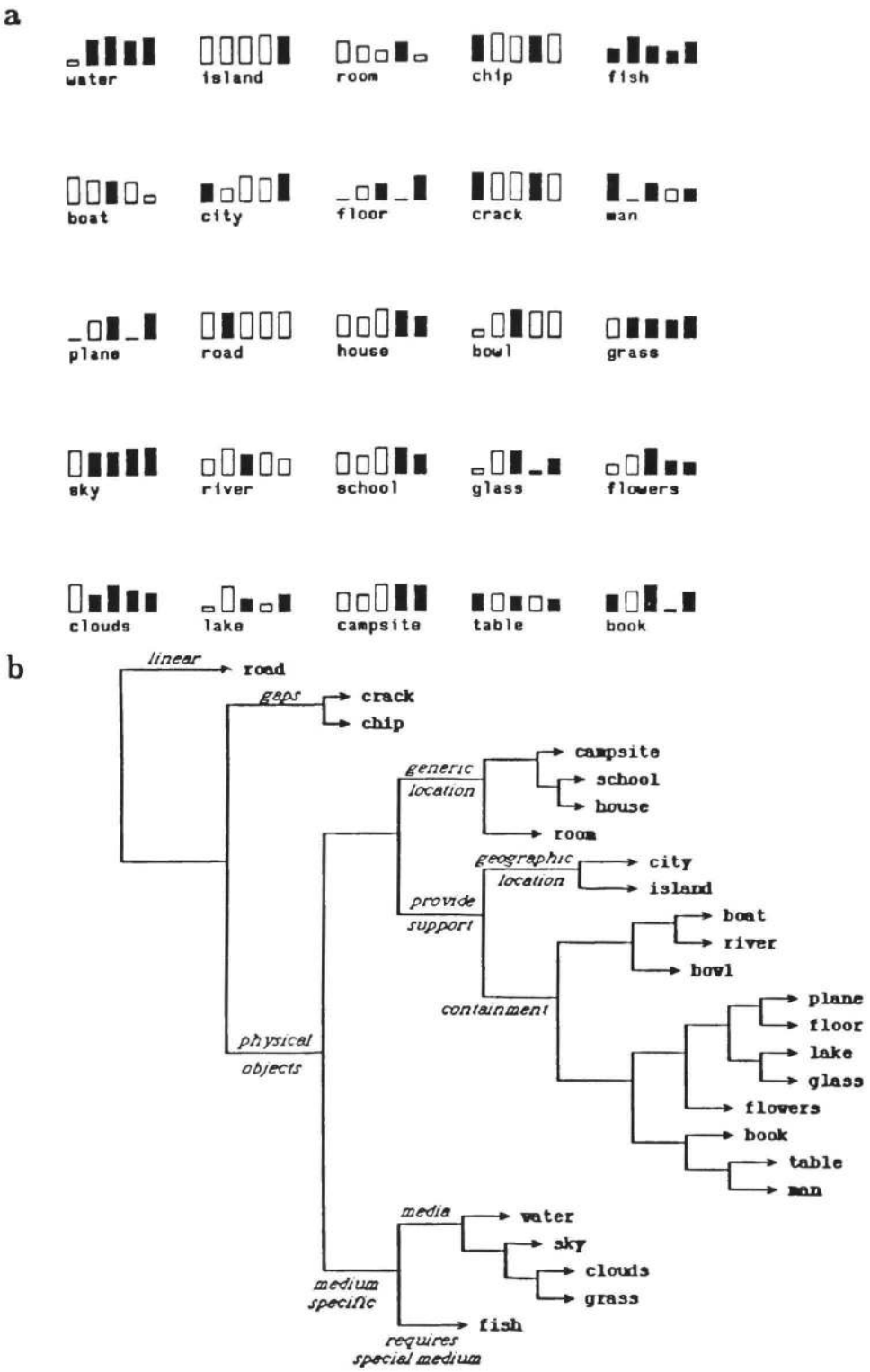


Figure 3. Distributed representations of the nouns. In order to accomplish the task successfully, the learning procedure was forced under this architecture to "discover" suitable distributed representations for the nouns. (a) the noun representations; (b) the tree represents the annotated results of a cluster analysis on these representations.

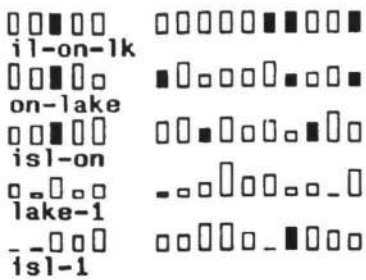


Figure 4. Context Sensitivity. The network's output in response to each stimulus is different. The response to each word alone is different from that given by one noun and a preposition, and from the response to the full phrase.

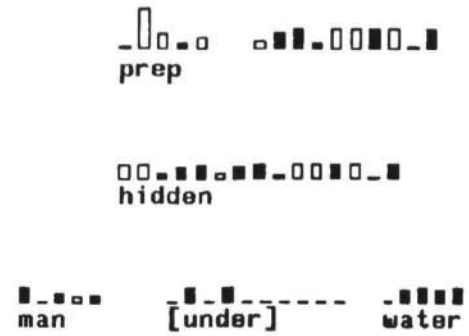


Figure 5. Network processing of the input *man [under] water*. The response of each layer in the network to the input is depicted. Note that the output units for both the preposition *under* and the preposition *in* are weakly activated.

The trained network was also run on a stimulus set of thirty-three novel phrases. Twenty-three were interpreted correctly. One of the more interesting results (Figure 5) was the response to the phrase *man [under] water*, where the brackets denote the appropriate semantic representation for *under* ("N2 over N1" and "N1 embedded in N2"). When the network was given N1, N2, and SP, the preposition chosen by the network was *under*. In addition, however, another preposition received activation (although less activation than *under*): *in*. Some of the results that were different from the correct response were plausible interpretations. For example, the phrase *grass on water* evokes either of two images -- *grass floating on water*, or *grass standing on the edge of the water*. The network, when given N1, N2, and LP gave the semantic units for the latter relationship, which initially surprised us, since we had not considered that interpretation before.

5. Discussion

The network was able to solve both the encoding and decoding task with negligible error. To solve the problem, reasonable meanings for the prepositions alone, and reasonable distributed representations for the nouns were developed. Because of this, the network was able to reasonably handle many of the novel inputs. Perhaps the most interesting aspect of this network lies in the fact that it is used to express the *semantic content* of any of a number of locative prepositions in a variety of contexts. We plan to explore the following conjecture: This semantic representation is language independent, therefore the mapping of semantic content to preposition is learned in both directions (i.e. prepositions can be *decoded* into semantics and semantics can be *encoded* into prepositions), semantic representations can be used to translate locative prepositions from one language to another.

Thus, we plan to explore the potential application of this work to machine translation. Other possible directions for extending this work include [1] increasing the scale of the model system (both in terms of the words used and the number of input patterns), [2] alternative architectures that can be used for variations on the encoding and decoding task, including metaphorical uses of prepositions, and [3] extending the uses of the prepositions beyond concrete spatial relationships to more metaphorical roles.

A Computational Model of Syntactic Ambiguity as a Lexical Process[1]

Curt Burgess
Department of Psychology
University of Rochester

S.C. Hollbach
Department of Computer Science
University of Rochester

Introduction

The use of semantic information in language comprehension is a matter of controversy. We briefly review Frazier's modular serial parser and Altmann and Steedman's conceptually driven parser. We propose a third model, based on the work of Tanenhaus and Carlson [1988],[Tanenhaus *et al.* 1987], that accounts for the conflicting results of these two models. We have built a connectionist implementation of our model, and, based on the behaviour of the network, make a number of empirical predictions.

The Controversy Defined

The work of Frazier and her colleagues ([Frazier and Rayner 1982]; [Ferreira and Clifton 1986]) is probably best known modular parser. They present considerable evidence that the parser uses a Minimal Attachment strategy. Minimal Attachment is used to explain the Bever sentences, e.g., *The horse raced past the barn fell.* The strategy is straightforward: words are attached to the phrase marker using the fewest possible syntactic nodes permissable by the syntactic rules. Frazier and Rayner [1982] have found that Non-minimal Attachment sentences take longer to read than Minimal Attachment sentences and that eye fixations in the disambiguating region of the sentences were of longer duration than elsewhere in the sentence.

It is possible, however, that a contextual manipulation could override the Minimal Attachment preference. Ferreira and Clifton [1986] consider this possibility. Their first experiment manipulates the animacy of the first noun phrase. They reasoned that the implausibility of an inanimate agent with a verb that is associated with an animate agent would be sufficient to eliminate the garden path in the reduced condition. For example, consider the following two sentences:

1a. The defendant examined by the lawyer turned out to be unreliable.
1b. The evidence examined by the lawyer turned out to be unreliable.

They found that animacy of the first noun phrase had no effect on reading times at either the *by* phrase or the second verb. However, eye fixations were longer at the first verb when it followed an inanimate NP, suggesting that eye fixations are a valid measure of semantic anomaly.[2] It seems, then, that the Minimal Attachment preference results in a garden path regardless of the animacy information in the first NP.

Ferreira and Clifton conduct another experiment in which they precede relative clause ambiguities such as *The editor played the tape agreed the story was a big one* with a paragraph that biases the interpretation of the sentence toward either a Minimal Attachment

[1] We want to thank Mike Tanenhaus and Bob Peterson for their encouragement, patience and many helpful suggestions. Send correspondence to Curt Burgess, Psychology Dept, University of Rochester, Rochester, NY 14627 [email: RCBU@UORDBV.BITNET]

[2] Apparently eye fixations reflect semantic as well as syntactic anomaly. It becomes less than completely clear what the linkage is between eye fixations and particular types of psycholinguistic processing.

or Non-minimal Attachment interpretation. The results of this experiment suggest that a preceding contextual paragraph with a Non-minimal Attachment bias is unable to influence the Minimal Attachment preference and that the contextual information is used solely to recompute the structural aspects of the sentences.

While the Frazier and Rayner [1982] and Ferreira and Clifton [1986] results appear quite convincing, this structural approach is not without criticism. Altmann and Steedman [in press] propose that sentences vary not only in structural complexity, but in presuppositional complexity. When a critical sentence in an experiment is encountered, its' relative clauses and prepositional phrases presuppose that a specific set of entities has been introduced by the context. Their claim is that context is ineffectual in the Ferreira and Clifton experiment simply because the context is impoverished and does not set up the necessary presuppositions. When Altmann and Steedman [in press] constructed Non-minimal Attachment supporting paragraphs that explicitly introduce each entity in the test sentence, they did not find the usual garden path effect. Altmann and Steedman's results present a serious challenge to a simple structural explanation such as offered by Frazier and Rayner or Ferreira and Clifton. It appears that lexical or semantic representations may have a substantial role to play in the structural organization of a sentence.

Thematic Roles and Semantic Ambiguity

These apparently conflicting results can be reconciled with a lexical access model of syntactic ambiguity. According to the lexical access model, Ferreira and Clifton's failure to find an animacy effect (e.g., *The evidence examined by the* ...) lies in the lexical representation of the verb. In their first experiment they compare sentences in which the first NP is either animate or inanimate. At the *by* phrase in such a sentence, the verb is shown to be unambiguously a past participle (in the reduced version). The local indeterminacy of the verb (i.e., past tense/past participle) is a morphological ambiguity with each verb form having its own thematic role mapping. The simple past tense tends to be more frequent than the past participle in most verbs. Since the two tenses of the verb are incompatible, the less frequent past participle is inhibited by the more frequent past tense. For example, compare

2a. The child bought the dog a leash. 2b. The child bought the dog was pleased.

In 2a, the preferred tense of *bought* is used and the two post-verbal NPs fill thematic roles that are subcatagorized by the verb. No garden path is experienced in this sentence. However, in 2b the reader is not aware that the past participle form is necessary for comprehension until the primary verb *was* is encountered. In our lexical model, both the past tense and the past participle are activated in parallel, but the less preferred form is quickly inhibited. As sentence processing continues it becomes increasingly difficult to reactivate the inhibited form. If, later in the sentence, the inhibited form of the verb is required, a garden path will occur.

Now, the results of Altmann and Steedman suggest that a garden path can be avoided under the right contextual circumstances.[3] We agree and suggest the following mechanism

[3] At present, we feel compelled to reject Ferreira and Clifton's argument that an inanimate NP is unable to propose the initial parse. There are at least two problems with their stimuli in experiment 1. Almost half of the inamimate subjects can act as agent (e.g., *The car towed* ...) and a number of the sentences

for the contextual effect: the less preferred past participle form of the verb is selected by the discourse model as it is made available by bottom-up lexical processes. The contextually inappropriate form is then inhibited. This understanding of how context can serve to select the appropriate verb form is important, although we have not yet implemented this in our model.

The model we propose, namely multiple activation of the verb forms followed by contextual selection, parallels the research on lexical ambiguity resolution. In isolation, access of ambiguous word meanings is a frequency coded, multiple access process [Burgess and Simpson 1988a, Simpson and Burgess 1985]. Multiple access obtains even in a sententially biasing context [Seidenberg *et al.* 1982, Burgess *et al.* in press]. The notion of parallel bottom-up activation has proved to be a powerful explanatory tool in lexical processing. Compatible lexical representations remain activated in such a model, while incompatable representations compete with each other based on the entities' resting activation level and previous exposure to related or associated items.[4] In our model, the garden path occurs as a function of the additional time and processing required to inhibit the inappropriate morphological verb form and to access the appropriate one.

In addition to verb forms, thematic roles play an important role in this model as well. Earlier research has shown that the meaning of a verb can be distinguished from its thematic roles. Tanenhaus, Burgess, D'Zmura and Carlson [1987] showed that a garden path occurs with a less preferred verb sense ambiguity (3a), but not with the preferred sense (3b). In contrast, no garden path occurs with either the less preferred (3c) or more preferred (3d) thematic role ambiguity.

3a. Bob passed the test to his friend.
3b. Bob passed the test to his surprise.
3c. Cathy threw the cat off the bed.
3d. Cathy threw the cat some Friskies.

Thematic roles behave differently than sense ambiguities. All thematic roles associated with a verb are activated in parallel at the point the verb is retreived and are provisionally assigned to verb arguments as soon as they are activated. Roles that are subcategorized by the verb remain activated even if unfilled and thus allow for rapid integration of concepts into the discourse model. For this reason they are an important entity for rapidly resolving local ambiguity.

The Connectionist Implementation

We have built a connectionist network that embodies the lexical model of syntactic ambiguity on the Rochester Connectionist Simulator [Goddard 1987]. The network is displayed in iconic form on a Sun Workstation, as depicted in Figure 1. The implementation adheres to the localist style of connectionist models, which is to say that the network is highly structured and each node has a precise semantics. Following Cottrell [1985], we adopt a form of exploded case roles, with a distinct node for each element of the cross product of thematic roles and syntactic position. For inner cases such as 'theme' there can be only one syntactic unit that plays this role, hence there are inhibitory links between all competing syntactic

continue the ambiguity beyond the *by* phrase. We've benefitted from discussions with Mike Tanenhaus, Susan Garnsey, Tom Bever and John Trueswell concerning this.

[4]This view can be contrasted with that of Marslen-Wilson and cohorts [Marslen-Wilson and Tyler 1987] in which top-down processing provides for selective access of lexical information.

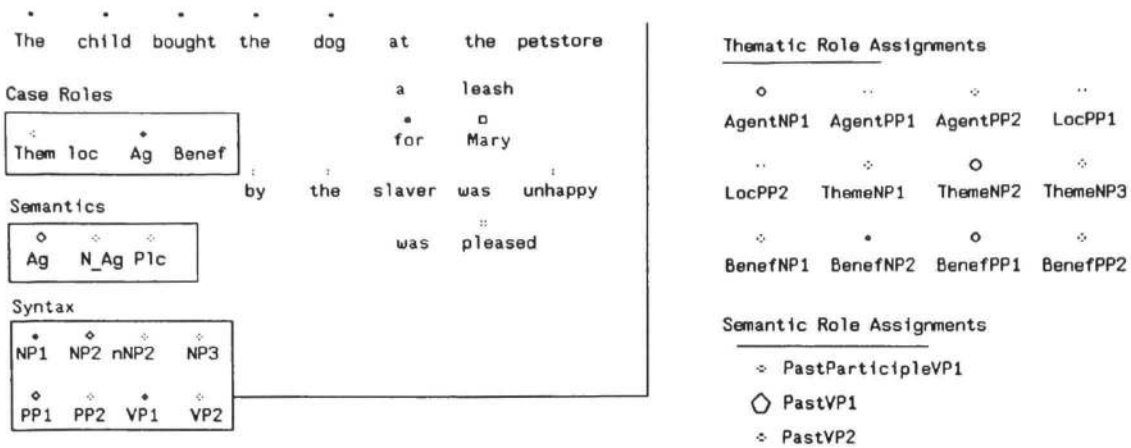


Figure 1: Visual display panel for the network. The input sentence 'The child bought the dog for Mary' is shown 10 iterations after sentence completion.

assignments. Moreover, a given syntactic unit can only take on one interpretation, hence there are inhibitory links between all competing thematic interpretations.

There are three distinct node types in the network: input nodes, role assignment nodes, and 'other' nodes. The activation function for 'other' nodes is simply to set the output of the node to the sum of all inputs:

$$a_t = \Sigma I_t,$$

where a_t is the activation level at time t and I_t is the input vector. Input nodes (or lexical item nodes) ignore their inputs, as the user keys activation in by hand for these nodes. The effect of reading a word in a sentence is simulated by the user setting the activation level of the relevant input node to 1.0. Activation of this node decays thereafter according to the recurrence relation:

$$a_t = a_{t-1} - \frac{1}{a_{t-1}}$$

In order to enter a stimulus sentence, the user keys in each word in turn, running the simulation a few steps between each word. The gradual decay of word activation means that by the time the last word of the sentence has been keyed in, the first word is only marginally active (see Figure 1). Thus any assumptions that have been made early on in sentence processing take a long time to reverse when invalidated by later constraints. The activation function for role assignment nodes incorporates a certain amount of inertia. If the potential of the unit is high, a relatively low weight is accorded to the input signal, while if the potential is low, the input signal has a larger impact on the updated activation level. This is accomplished by dividing the input signal (I_t) by the prior activation before adding the whole to the current activation:

$$a_t = a_{t-1} + \frac{cI_t}{a_{t-1} + k}, \quad c = 5, \quad k = 10.$$

Interpretations are advanced as the sentence progresses. A schematic of the some of the important features of network structure appears in Figure 2. As shown in Figure 2, any number of thematic role assignments are possible after the first noun phrase (NP1), so all are advanced with equal confidence. For the verb 'bought' (VP1) there are two possible interpretations: the more likely is the past tense, the less likely the past participle. Thus the past tense recieves preferential activation, and, as the two interpretations are in direct

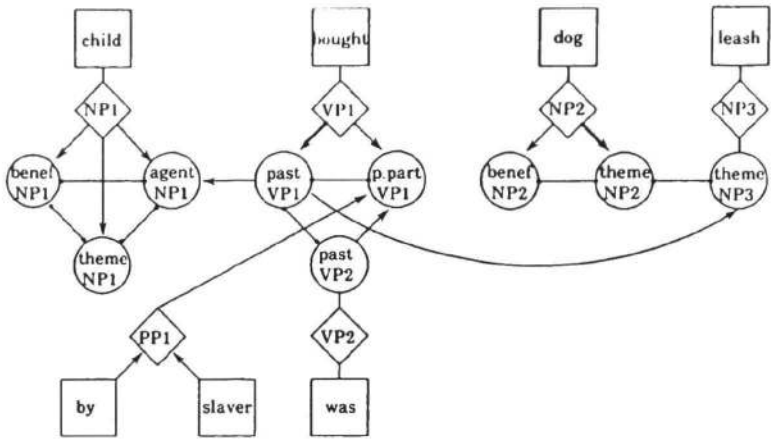


Figure 2: Rough schematic of network structure. Not all nodes and links are shown. Bold arrows indicate preferential weighting on one of multiple outputs. Input nodes appear as squares, role nodes as circles, and other nodes as diamonds.

competition, the second alternative tends to decrease in likelihood as the sentence processing continues.

The first noun phrase encountered after the verb (NP2) is ambiguous itself, but constrains the role of NP1 to either *agent* (if the verb is in the past tense), or *beneficiary* (if the past participle). The ambiguity of NP2 is resolved by the subsequent syntactic unit: if NP3 occurs, then the thematic role assignments are *benefNP2* and *themeNP3*; but if a prepositional phrase (PP1) is encountered, the assignment becomes *themeNP2*, with the role of PP1 depending on its composition.

If a past tense verb (VP2) is now encountered, the only possible interpretation for VP1 becomes the past participle. By this time, however, activation on the *pastVP1* role assignment node will have reached significant proportions, and will only respond slowly to the attempt to inhibit it, so the reversal of the semantic ambiguity is a gradual process. Upon successful reversal, the interpretation of NP1 is altered from *agent* to *beneficiary*.

One syntactic cue that will tend to increase the confidence of the past participle interpretation in mid-sentence is the occurrence of a prepositional phrase immediately following the verb, since *to buy* is transitive.

All of the above constraints are tabulated in Figure 3, and timing results of running the network on the sample stimuli are given. There are three classes of sentence: a straightforward past tense, understood quickly, an unusual but clearly signalled past participle construction, understood after a moderate delay, and a misleading garden path involving a past participle, understood only after a lengthly reconsideration. Timing results reflect the time elapsed from reaching the end of the sentence until each of the correct thematic and semantic role assignment nodes reaches an activation level of at least 0.5.

Conclusions

We have described a model that accounts for a variety of results without appealing to Minimal Attachment strategies. The parser differs from that of Frazier in several respects. It utilizes parallel activation of morphological verb forms that makes available more than one parse option that can then be selected by context. The model accounts for the Non-minimal Attachment garden path as a function of the frequency asymmetry of the verb forms. This

NP_{agent}	'bought'$_{past}$	NP_{theme}	{PPs}	timing
The child	bought	a dog.		46
The child	bought	a dog	at the petstore.	47
The child	bought	a dog	for Mary.	48

NP_{agent}	'bought'$_{past}$	NP_{benef}	NP_{theme}	
The child	bought	the dog	a leash.	46

NP_{theme}	'bought'$_{p.part}$	PPs	$VP_{past}\ldots$	
The child	bought	by the slaver	was sad.	62

NP_{benef}	'bought'$_{p.part}$	NP_{theme}	$VP_{past}\ldots$	
The child	bought	a dog	was pleased.	109

Figure 3: Allowable syntactic forms of sentences using 'bought', and network times for successful comprehension.

would predict that the magnitude of the garden path would be related to this asymmetry, assuming that context is held constant. This model also predicts that preceding context can affect the parse by selecting, after lexical access, the appropriate form. As currently envisioned, context does not selectively affect access but merely functions as a post-access selection mechanism, thus allowing the model to be modular in nature.

The model is closer in spirit to how Altmann and Steedman [in press] have conceptualized sentence comprehension. Their contextual presuppositions serve to select the appropriate morphological verb form, thus reducing the garden path. The current proposal adds to Altmann and Steedman's model in that we have specified the mechanism by which context exerts its influence. The immediate empirical questions to be answered include tracing the timecourse of the activation and inhibition of the morphological verb forms and evaluating the locus of the contextual effect.

Neuropsychological Processes

Placing syntactic processing in the lexicon raises some intruiging questions concerning the locus of processing in garden path recovery. Burgess and Simpson [1988a] found that subordinate associates to ambiguous words are rapidly inhibited in the left hemisphere, but maintain activation over a relatively long period of time in the right hemisphere. Dominant associates maintain activation over time, but appear to decay in the right hemisphere. It is possible that these cerebral asymmetries may aid in semantic garden path recovery [Burgess and Simpson 1988b]. That is, if a sentence was not disambiguated until after the ambiguous word and the subordinate meaning were needed, the language processor would just access this meaning from the right hemisphere. This precludes the need to reactivate the meaning in the left hemisphere. If the syntactic lexical effects discussed above are bilateral in their representation, then one might expect a role for the right hemisphere in syntactic garden path recovery. While this is speculative, particularly since syntax is generally thought to be the domain of left hemisphere processing, Schneiderman and Saddy [in press] have reported that right brain damaged patients have difficulty inserting words into a sentence when the

insertion requires a syntactic reassignment of a word.

Summary

This model proposes a lexically located syntactic processor that makes available multiple morphological verb forms in parallel. It is modular in architecture, but differs from the modular serial parser proposed by Frazier. The model could easily be extended to allow for post-access contextual selection of the appropriate verb form, thus avoiding the Minimal Attachment garden path. The model suggests that the nature of this contextual effect, as detailed by Altmann and Steedman, is lexical selection. Several empirical issues involving frequency asymmetries of the verb forms, the speed of retrieval and inhibition, and interactions with context were discussed. This research is necessary in order to determine the correctness of the present implementation.

References

[Altman and Steedman in press] Gerry Altman and Mark Steedman, "Interaction with Context during Human Sentence Processing", *Cognition*, in press.

[Burgess and Simpson 1988a] Curt Burgess and Greg B. Simpson, "Cerebral Hemispheric Mechanisms in the Retrieval of Ambiguous Word Meanings", *Brain and Language*, 33:86–103, 1988.

[Burgess and Simpson 1988b] Curt Burgess and Greg B. Simpson, "Neuropsychology of Lexical Ambiguity Resolution: Contribution of Divided Visual Field Studies", *Lexical Ambiguity Resolution in the Comprehension of Human Language*, Los Altos, CA: Morgan Kaufmann, 1988.

[Burgess *et al.* in press] Curt Burgess, Michael Tanenhaus and Mark Seidenberg, "Context and lexical access: Implications of nonword interference for lexical ambiguity resolution", *Journal of Experimental Psychology: Learning, Memory and Cognition*, in press.

[Cottrell 1985] Garrison W. Cottrell, "A Connectionist Approach to Word Sense Disambiguation", Technical Report 154, Department of Computer Science, University of Rochester, May 1985.

[Ferreira and Clifton 1986] Fernanda Ferreira and Charles Clifton, "Independence of Syntactic Processing", *Journal of Memory and Language*, (25):348–368, 1986.

[Frazier and Rayner 1982] Lyn Frazier and Keith Rayner, "Making and Correcting Errors during Sentence Comprehension: Eye Movements in the Analysis of Structurally Ambiguous Sentences", *Cognitive Psychology*, (14):178–210, 1982.

[from Right Hemisphere Damage in press] A Linguistic Deficit Resulting from Right Hemisphere Damage, "E. Schneiderman and J.D. Saddy", *Brain and Language*, in press.

[Goddard 1987] Nigel Goddard, "The Rochester Connectionist Simulator User Manual", Technical Report, Computer Science Department, University of Rochester, April 1987.

[Marslen-Wilson and Tyler 1987] W. Marslen-Wilson and L. Tyler, "Against Modularity", *Modularity in Knowledge Representation and Natural Language Understanding*, pages 37–62, Bradford Books, MIT Press, 1987.

[Seidenberg *et al.* 1982] Mark Seidenberg, Michael Tanenhaus, J.M. Leiman and M. Bienkowski, "Automatic Access of the Meanings of Ambiguous Words in Context: some Limitations of Knowledge-based processing", *Cognitive Psychology*, 14:489–537, 1982.

[Simpson and Burgess 1985] Greg B. Simpson and Curt Burgess, "Activation and Selection Processes in the Recognition of Ambiguous Words", *Journal of Experimental Psychology: Human Perception and Performance*, 11:28–39, 1985.

[Tanenhaus and Carlson 1988] Michael K. Tanenhaus and Greg Carlson, "Lexical Structure and Language Comprehension", *Lexical Representation and Process*, Cambridge, MA: MIT Press, 1988.

[Tanenhaus *et al.* 1987] Michael K. Tanenhaus, Curt Burgess, Susan H. D'Zmura and Greg Carlson, "Thematic Roles in Language Processing", *Proceedings of the 1987 Conference of the Cognitive Science Society*, 1987.

A PARALLEL MODEL FOR ADULT SENTENCE PROCESSING[1]

ROBIN CLARK AND EDWARD GIBSON

CARNEGIE MELLON UNIVERSITY

1. INTRODUCTION

Serial deterministic models of human sentence processing (e.g. Frazier, 1978; Frazier & Fodor, 1978; Marcus, 1980; Berwick, 1985) have had a great intuitive appeal, especially when contrasted with unlimited parallel processing. A serial deterministic parser, for example, provides a straightforward account of why people often fail to perceive ambiguity and why they report a strongly favoured interpretation for ambiguous sentences. Serial deterministic processing, furthermore, gives a very plausible account of how people are misled into a garden-path parse for certain strings:

(1) `The horse raced past the barn fell.`

By virtue of returning all possible grammatical parses for an input sentence, an unlimited parallel parser, by itself, gives us no reason to prefer one parse over another, and, prima facie, such a model fails to account for the preferred readings of ambiguous input. Because of its computational power, a parallel parser cannot be misled, since the non-preferred analysis of garden-path input can be carried along by the parallelism. Hence, a fully parallel parser provides no obvious account of garden-path phenomena.

Parallel models of processing, on the other hand, can provide a simple account of relative processing load.[2] When the parser encounters ambiguous input, the number of hypotheses entertained by the parser will increase and more computational resources (memory and time measured in number of operations) will be used. Since a deterministic model is committed to a single hypothesis at all times (Marcus, 1980; Berwick, 1985), the computational resources used by the parser should remain virtually constant. A deterministic model, therefore, has no obvious way of modeling variations of processing load with respect to ambiguous input (Gorrell, 1986).

Constrained parallel processing may provide an account of garden-path effects and preferred readings for ambiguous input, on the one hand, and variations in the relative complexity of processing, on the other. As long as a parallel model is limited so that it obtains garden-path and other classic psycholinguistic effects generally associated with serial models, it is a valid model. Until recently, however, serial models have been generally preferred to parallel ones, since it was not obvious how to constrain the power of a fully parallel model.

It has been observed in a number of studies that word recognition is parallel (Swinney, 1979; Tanenhaus, Leiman & Seidenberg, 1979; Seidenberg, Waters, Sanders & Langer, 1984). Since syntactic processing makes use of the results obtained from word recognition, a deterministic syntactic processor may select only one lexical item from this list. It is unclear how to constrain the lexical decision task imposed on the parser without recourse to devices such as look-ahead or multiple passes over the input string (cf., the "attention shifts" of Marcus, 1980). Evidence indicates that lexical selection is not sensitive to pragmatic knowledge (Swinney, 1979). Following Fodor (1983),

[1]We would like to thank Rick Kazman for his comments on an earlier draft of this paper.

[2]On variations in relative complexity of processing, see Fodor, Garrett & Bever, 1968; Holmes & Forster, 1972; Hakes, 1972; Rayner, Carlson & Frazier, 1983; Frazier & Rayner, 1987; Shapiro, Zurif & Grimshaw, 1987, among others.

we assume that syntactic processing is "informationally encapsulated"; that is, the syntactic processor does not make appeal to extra-linguistic pragmatic knowledge. Furthermore, we assume that syntactic processing is automatic in that it is fast and not subject to conscious manipulation. A parallel syntactic processor offers great uniformity between lexical recognition and sentence processing. Parallel syntactic processing may also illuminate the difficult task of selecting the syntactically appropriate reading of ambiguous lexical items without appeal to problematic constructs like look-ahead and multiple passes by the parser. Recently, psycholinguists have reconsidered parallelism in the context of sentence processing (Kurtzman, 1985; Gorrell, 1986; Carlson & Tanenhaus, 1986), although few explicit parallel models have been formulated. We will describe a parallel algorithm for sentence processing which is compatible with recent research in psycholinguistics in that it produces well-known effects; furthermore, since the model is explicitly formalized, its predictions are readily subject to empirical disconfirmation.

In this paper we present Constrained Parallel Parser (CPP), a parser based on the principles of Government-Binding Theory (Chomsky, 1981; Chomsky, 1986); crucially, CPP has no separate grammar rule module containing language-particular rules. Unlike many psycholinguistic models, CPP has been implemented; it exists as a CommonLISP program. In the CPP model, distinct tree structures correspond to different argument structures of a given lexical entry. When a word is input, these representations are built and placed in the *buffer*, a one cell data structure that holds a parallel list of these representations. Hence, the complexity of the parse will be contingent on the number of argument structures associated with a lexical item (cf., Shapiro, Zurif & Grimshaw, 1987). CPP contains a second data structure, the *stack*, that is of the same structure as the buffer, but may be more than one cell deep. The parser builds trees in parallel based on possible attachments made between the buffer and the top of the stack. In addition, the attachments are limited by constraints on the algorithm and constraints on the resulting representations. Both types of constraints can be shown to cause garden-path effects (see Clark, 1987; Gibson, 1987).

2. LEXICAL ENTRIES FOR CPP

A lexical entry accessed by CPP consists in part of a *theta-grid*. A theta-grid is an unordered list of *theta structures*. Each theta structure consists of a thematic role and associated subcategorization information. One theta structure in a theta-grid may be marked as *indirect* to indicate that it must be assigned to the subject of the phrase. For example, the word *shout* might have the following theta-grid:[3]

```
((Subcat = NOUN, Thematic-Role = AGENT, INDIRECT)
 (Subcat = PREP, Thematic-Role = GOAL)
 (Subcat = COMP, Thematic-Role = PROPOSITION))
```

When the word *shout* [4] is encountered in an input phrase, the thematic role *agent* will be assigned to its subject, as long as this subject is a noun phrase. The direct thematic roles *goal* and *proposition* will be assigned to prepositional and complementizer phrases respectively, as long as each is present. Since the order of theta structures in a theta-grid is not relevant to its use in parsing, the above theta-grid for *shout* will be sufficient to parse both sentences (2) and (3).

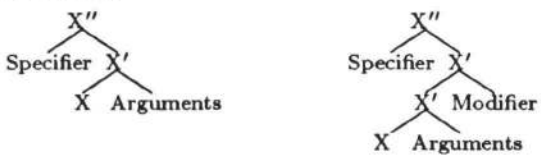
[3] In a more complete theory, a syntactic category would be determined from the thematic role.

[4] In the current system, morphological variation is only covered by explicit listing of lexical entries. A morphological analysis component is proposed to extend the current system.

(2) The man shouts to the woman that Ernie sees the rock.
(3) The man shouts that Ernie sees the rock to the woman.

3. THE CPP ALGORITHM

The CPP model assumes $\overline{X}$ Theory as present in Chomsky (1986). Each tree structure must have a head and each head must have a maximal projection. These principles interact with other principles built into the parser, (*e.g.*, the θ–Criterion, the Extended Projection Principle, Case Theory) to determine the positions of arguments, specifiers and modifiers with respect to the head of a given structure. As a result, a specifier may only appear as a sister to the one-bar projection below a maximal projection, and the head must appear below the one-bar projection along with its arguments. For example, the structure for categories in English is shown on the left below, with a modifier attachment on the left below.



The Constrained Parallel Parser is a head-driven parser that builds structure by making attachments between the buffer and the top of the stack. For each lexical entry, a maximal projection of that entry is placed in the buffer. The *buffer* is a one cell data structure that contains a set of tree structures, each of which represents the same segment of the input string. The data structure *stack* consists of a stack of buffer cells.

Since CPP has no separate grammar rule module containing language-specific rules, an attachment between a structure in the buffer and a structure on the stack is possible based the configuration of the given parser. This parser contains constants that are independent of the language being parsed along with parameters that depend on the language being parsed. For example, the fact that determiners, if they exist as an independent category in a given language, are attached in the specifier position of NP follows from Universal Grammar. Whether this attachment takes place from stack to buffer or from buffer to stack varies according to the type of attachment and language being considered, however. In English, specifier attachment takes place from stack to buffer, indicating that, in English, specifiers occur before the head. As a result, a parameter would be set in a parser for English that indicates specifier attachment occurs from stack to buffer. Arguments, on the other hand, are attached from buffer to stack in English, since English is head-first with respect to arguments. As with the case of Specifiers, this order is the result of setting a parameter which dictates the direction of the attachment of complements with respect to the head.

The formal CPP algorithm is given below, with parameters for attachment set to parse English.

1. (Initializations) Set the STACK to NIL. Set the BUFFER to NIL.
2. (Ending Condition) If the input string is finished and the BUFFER is empty then return the contents of the STACK and stop.
3. If the BUFFER is empty then create a maximal projection for each lexical entry corresponding to the next word in the input string, and put this list of maximal projections into the BUFFER.

4. Make all possible attachments between the STACK and the BUFFER, subject to the attachment constraints. Put the attached structures in the BUFFER. If no attachments are possible, then put the contents of the BUFFER on top of the STACK.
5. Go to 2.

Possible Attachments (parameterized for English):

- Argument Attachment: (BUFFER to STACK) If a structure B in the BUFFER is compatible with the lexical requirements of a structure A, on top of the STACK, then attach B to A as an argument.
- Specifier Attachment: (STACK to BUFFER) If a structure A, on top of the STACK, is compatible as a specifier of a structure B, in the BUFFER, then attach A to B as a specifier.
- Pre-Head Adjunct Attachment: (STACK to BUFFER) If a structure A, on top of the STACK, is compatible as a modifier of a structure B, in the BUFFER, then attach A to B as a modifier.
- Post-Head Adjunct Attachment: (BUFFER to STACK) If a structure B in the BUFFER is compatible as a modifier of a structure A, on top of the STACK, then attach B to A as an modifier.

Attachment Constraints:

- **Exclusive Attachment Constraint**: If an attachment is possible between two structures (one on the stack, one in the buffer), then it is made. All nodes in parallel that do not take part in attachment, either on the stack or in the buffer, are pruned.
- **Case Filter and θ-Criterion**[5]: After attachments between stack and buffer are completed, if a structure A directly receives a necessary property (*e.g.*, a thematic role, abstract Case for a lexical noun phrase), then prune all representations in which structure A appears but does not directly receive that property. For example, if a certain lexical noun phrase NP_1 receives Case in one representation, then all representations are pruned in which NP_1 does not receive Case.
- **Lexical Requirement Constraint**: If an attachment is possible that satisfies the lexical requirements of some head, then make that attachment and all others that also satisfy lexical requirements. If no such attachment is possible, then make any other possible attachments satisfying other constraints.

The Exclusive Attachment Constraint (EAC) guarantees that each of the parallel structures in a buffer cell represents the same segment of the input string. This uniformity permits the use of simple data structures. The EAC is extended into the Principles of Government-Binding theory to give the CPP versions of the Case Filter and θ-Criterion. The Lexical Requirement Constraint (LRC) is a parsing extension of the Projection Principle[6]. As a result of these constraints, garden-path and other psycholinguistic effects occur.

The CPP implementation is on a serial machine, so the parallelism must be mimicked. The parser still runs quite fast, averaging about one-tenth of a second per word on a Hewlett Packard 9000/350 with 8 megabytes of RAM. This speed can be partially attributed to the empirically observed fact that most parallelism dies very quickly using the CPP algorithm.

[5]The Case Filter states that a lexical noun phrase must receive abstract Case. The θ-Criterion states that all arguments must receive exactly one thematic role and that all thematic roles must be assigned.

[6]The Projection Principle states that all lexical requirements must be satisfied.

4. GARDEN-PATH EFFECTS AND CPP

To illustrate the algorithm in action, consider the garden-path sentence, sentence (4):

(4) **The man walked to the station ate the cake.**

This is a garden-path sentence because *walked to the station* is misanalyzed as matrix level verb phrase: to obtain a grammatical sentence, *walked to the station* must be analyzed as a reduced relative clause modifying the noun phrase *the man.*

The parse begins with the placement of a maximal projection for the determiner *the* in the buffer. Since there is nothing on the stack, no attachments can be made, and the determiner phrase simply moves to the stack. The second word, *man*, is then read from the input string. Since *man* has both noun and verb entries in the lexicon, a maximal projection for each reading enters the buffer, as shown below:

BUFFER: ($[_{N''}$ $[_{N'}$ $[_{N}$ *man*]]]
$[_{V''}$ $[_{V'}$ $[_{V}$ *man*]]])
STACK: (($[_{Det''}$ $[_{Det'}$ $[_{Det}$ *the*]]]))

We note that, at this point, the relative processing load has increased due to the ambiguity of *man.* Now that both the stack and buffer are non-empty, attachments may be tried. Argument attachment fails, since the structure on top of the stack, the determiner phrase representing *the*, has no lexical requirements. Since a determiner cannot modify or be modified by either a noun or verb phrase, both pre- and post-head modifier attachments fail. Specifier attachment fails between the determiner and the verb, but succeeds between the determiner and the noun, since a determiner is a possible specifier for a noun phrase.[7] The noun phrase resulting from attachment is then placed in the buffer. Since the verb phrase reading of *man* did not take place in the attachment, it is pruned from the parse by the Exclusive Attachment Constraint. At this point, the complexity of the parse decreases, since one representation has been eliminated. The contents of the buffer are then moved onto the stack and maximal projections for the next input word, *walked*, are placed in the buffer. These projections consist of a verb phrase representing the passive participle *walked*, as well as an Infl phrase representing the tensed verb *walked.*[8]

BUFFER: ($[_{Infl''}$ *walked*]
$[_{V''}$ *walked*]
STACK: (($[_{N''}$ *the man*]))

Argument attachment fails, since the noun phrase *the man* has no lexical requirements. Specifier attachment succeeds between the Infl phrase and the noun phrase, as well as between the verb phrase and the noun phrase. Modifier attachment also succeeds between the noun phrase in the buffer and the verb phrase on the stack. Since nominative Case is assigned to the NP specifier of the Infl phrase attachment, the Case Filter is activated. No Case is assigned to the NP *the man* in the small clause verb phrase reading or in the noun phrase with modifier reading, so these two

[7]This is presumably a theorem of Universal Grammar. That is, a determiner may be attached as the Specifier of a noun phrase. Thus, provided that a language has determiners, they will be attached as [Spec, N]. We assume, furthermore, that UG allows only this role for determiners; they cannot be modifiers of VP for example.

[8]The category *Infl* contains inflection information. An Infl phrase is automatically built for any tensed verb, since tense resides in Infl in the GB framework.

representations are pruned. As a result, the reduced relative clause reading of the phrase *walked to the station* is ignored, which eventually leads to the garden-path effect. Maximal projections for the word *to* now enter the buffer: a prepositional phrase and Infl phrase. Both attach as modifiers to the matrix verb phrase. The determiner *the* then enters the buffer. No attachments are possible at this point, so the contents of the buffer are pushed onto the stack. A maximal projection for the noun *station* is then placed in the buffer and the determiner on top of the stack attaches to it. The state of the parse at this point is as follows:

BUFFER: ($[_{Infl''}$ $[_{N''}$ *the man*] $[_{Infl'}$ $[_{V''}$ $[_{V'}$ $[_{V'}$ *walked* $[_{P''}$ *to*]]]]]]
$[_{Infl''}$ $[_{N''}$ *the man*] $[_{Infl'}$ $[_{V''}$ $[_{V'}$ $[_{V'}$ *walked* $[_{Comp''}$ $[_{Infl''}$ *to*]]]]]]]
STACK: (($[_{N''}$ *ihe station*]))

Only argument attachment succeeds for the prepositional reading of *to*, while no attachment is possible for the Infl reading of *to*. The Exclusive Attachment Constraint therefore prunes the structure containing the embedded Infl reading.

The tensed Infl phrase *ate* now enters the buffer. No attachments are possible between the two Infl phrases and the Extended Projection Principle[9] is violated. The garden-path effect follows naturally from general constraints on the parallelism displayed by the parser. These same constraints are partially responsible for the observed speed of the parser.

5. CONCLUSIONS

We have described a parallel parsing model that, like serial deterministic models, obtains garden-path effects. The design of the parser follows from current work in syntactic theory. The representations posited by the parser must obey certain constraints (the Case Filter, θ-Criterion, etc.). Furthermore, in keeping with the spirit of recent work in Government-Binding theory (Stowell, 1981; Chomsky, 1985), the parser makes no use of language-particular grammar rules. The absence of language-particular rules, in conjunction with constraints on parallelism are responsible for much of the speed of the parser. Given the parser's ability to replicate phenomena like garden-path effects, we feel that research along these lines can do much to illuminate the relationship between knowledge and its use. Finally, we note that the Constrained Parallel Parser is a genuinely parallel parser. Unlimited parallel parsers cannot obtain garden-path effects, because of their inability to err. The CPP model, since it is severely constrained, does not suffer this defect. Hence, psycholinguistic theory, while correct in abandoning unconstrained parallelism, stands to profit from the study of constrained parallel algorithms. Finally, the CPP algorithm provides a non-trivial alternative to standard methods of parsing that make use of charts, networks or case-frames; the algorithm has the potential of yielding an interesting formal basis for the empirical study of adult sentence processing.

REFERENCES

Berwick, R. (1985), *The Acquisition of Syntactic Knowledge*, MIT Press, Cambridge, MA.

Carlson, G.N., & Tanenhaus, M.K. (1987), *Thematic Roles and Language Comprehension*, University of Rochester manuscript.

[9]The Extended Projection Principle (Chomsky, 1981) states that all lexical requirements must be satisfied and that all Infl phrases must have subjects.

Chomsky, N. (1981), *Lectures on Government and Binding*, Foris, Dordrecht, The Netherlands.

Chomsky, N. (1985), *Knowledge of Language: Its Nature, Origin and Use*, Praeger Publishers, New York, NY.

Chomsky, N. (1986), *Barriers*, Linguistic Inquiry Monograph 13, MIT Press, Cambridge, MA.

Clark, R. (1987), *Rules and Parsing*, Talk presented at MIT.

Fodor, J.A. (1983), *Modularity of Mind*, MIT Press, Cambridge, MA.

Fodor, J.A., Garrett, M.F., & Bever, T.G. (1968), *Some Syntactic Determinants of Sentential Complexity*, Perception and Psychophysics 2, pp. 289-96.

Frazier, L., & Fodor, J.D. (1978), *The Sausage Machine: A New Two-stage Parsing Model*, Cognition 6, pp. 291–325.

Frazier, L. (1978), *On Comprehending Sentences: Syntactic Parsing Strategies*, University of Massachusetts Ph.D. dissertation. Frazier, L. (1985), *Syntactic Complexity*, from Dowty, Karttunen & Zwicky (eds.), *Natural Language Parsing*, Cambridge University Press.

Frazier, L. & Rayner, K. (1987), *Resolution of Syntactic Category Ambiguities: Eye Movements in Parsing Lexically Ambiguous Sentences*, Journal of Memory and Language 26.

Gibson, E.A.F. (1987), *Garden-Path Effects in a Parser with Parallel Architecture*, Eastern States Conference on Linguistics.

Gorrell, P.G. (1987), *Studies of Human Syntactic Processing: Ranked-Parallel versus Serial Models*, University of Connecticut Ph.D. dissertation.

Hakes, D.T. (1972), *Effects of Reducing Complement Constructions on Sentence Comprehension*, Journal of Verbal Learning and Verbal Behavior 11, pp. 278-86.

Holmes, V.M. & Forster, K.I. (1972), *Perceptual Complexity and Understanding Sentence Structure*, Journal of Verbal Learning and Verbal Behavior 11, pp. 148-56.

Jackendoff, R. (1977), *X-bar Syntax: A Study of Phrase Structure*, Linguistic Inquiry Monograph 2, MIT Press, Cambridge, MA.

Kurtzman, H. (1985), *Studies in Syntactic Ambiguity Resolution* MIT Ph.D. dissertation.

Marcus, M. (1980), *A Theory of Syntactic Recognition for Natural Language*, MIT Press, Cambridge, MA.

Rayner, K., Carlson, M., & Frazier, L. (1983), *The Interaction of Syntax and Semantics during Sentence Processing: Eye Movements in the Analysis of Semantically Biased Sentences*, Journal of Verbal Learning and Verbal Behavior, 22, pp 358-374.

Seidenberg, M.S., Waters, G.S., Sanders, M., & Langer, P. (1984), *Pre- and Post-Lexical Loci of Contextual Effects on Word Recognition*, Memory and Cognition, 12, 315-328.

Shapiro, L., Zurif, E. & Grimshaw, J. (1987), *Sentence Processing and the Mental Representation of Verbs*, Cognition 29.

Stowell, T. (1981), Origins of Phrase Structure, MIT Ph.D. dissertation.

Swinney, D.A. (1979), *Lexical Access during Sentence Comprehension: (Re)consideration of Context Effects*, Journal of Verbal Learning and Verbal Behavior, 18, pp 549-569.

Tanenhaus, M.K., Leiman, J.M., & Seidenberg, M.S. (1979), *Evidence for Multiple Stages in the Processing of Ambiguous Words in Syntactic Contexts* Journal of Verbal Learning and Verbal Behavior, 18, pp 427-440.

PARSING METACOMMUNICATION IN NATURAL LANGUAGE DIALOGUE TO UNDERSTAND INDIRECT REQUESTS[1]

David L. Sanford[2] & J. W. Roach

Department of Computer Science
Virginia Tech
Blacksburg, VA 24061

ABSTRACT

This paper reports on development of a natural language processing system based on human communication theory. Our system, DIALS (for DIALogue Structures), implements and extends the theory of metacommunication developed in the field of human speech communication. The theory of Dialogue Structures is based on research showing that the interpretation of conversation is enabled by metacommunications helpful in managing interactions and that indirect requests are usually patterns expressing relationships in the interaction rather than simply expressing the content of the request. As such, indirect requests are best interpreted by a semantic grammar expert at managing communciation, rather than a semantic grammar knowledgable on some specific task domain. Our system, based on this approach, correctly interprets all indirect requests from a corpus of 1500 requests transcribed from tape recordings with a combined total of over 80 minutes of continuous conversation of 27 dialogues between airline reservation agents and customers.

INTRODUCTION

We approach the programming of NLP systems from the perspective of human communication theory. We believe that the theory of metacommunication (Sanford & Roach, 1987a, 1987b, 1987c, 1988, in press) provides an approach to framing linguistic utterances. Indeed, we argue that the communicative behavior of the interactants is more important than task-specific knowledge for making such inferences. Our

---

[1] The research reported here was conducted at Virginia Tech and constitutes partial completion of the requirements for the Ph. D. of David L. Sanford at the University of Illinois.

[2] Currently at: Boeing Aerospace
Mail Stop: 82-58
P.O. Box 3999
Seattle, WA 98124

theory of Dialogue Structures currently does not cover the entire domain of interpersonal interaction. People with a history of interaction develop unique phrasings and interpretations, requiring representations of memory structures for interpersonal relationships that we have not developed yet. Therefore, we are focusing our initial research on contexts that involve strangers interacting. In particular, we choose to focus on interactions between airline reservation customers and agents, which involves utterances rich with requests. In this paper, we will explain how DIALS (our implementation of DIALogue Structures) identifies and uses metacommunicational knowledge to determine the meaning of indirect requests uttered in such contexts. We cannot explain everything about our theory in seven pages, so our explanation of the theory will be incomplete, at best.

INDIRECT REQUESTS

Grosz (1980) and Allen (1983) define indirect requests as requests in which the surface utterance appears to be asking for a yes/no response, while the underlying intention is to get something else. We consider indirect requests to be conventionalized phrasings of pleading "imperative force," a metacommunicational expression used to manage interaction rather than a source of task-specific information. We divide this section into two parts: we first analyze how people use indirect requests and then we explain how DIALS handles indirect requests.

How People Use Indirect Requests

There are two possible functions of requests worded indirectly: first, they are **not** truly indirect and are asking for a yes/no response; second, they **are** truly indirect and expect the respondent to do more. A communicator is confronted with deciding which is intended and why.

Knowledge Needed for Analyzing Indirect Requests

Content metaknowledge. When an "indirect request" is actually direct, i.e., when the requestor wants a yes/no response, metaknowledge about the task domain is needed to answer the question. First consider an example from Allen (1983): "Do you know when the Windsor train leaves?" How could a NLP system determine whether to answer this with a yes or no? It could examine its database, seeing if it can find a departure time for a train headed to Windsor. If it finds such an entry, then it answers "yes"; if not, it answers "no." Either way, it determines the embedded request and bases its answer on its success at fulfilling the

underlying request. But consider this strategy with an example from Grosz (1980): "Can you help me get a banana?" To answer, it tries to help get a banana. If it is successful at getting a banana, then it says "yes." But we began by assuming that the requestor simply wanted a yes/no response. To give such a response, this strategy requires that the embedded indirect request be fulfilled. The better strategy is to provide the system with metaknowledge about the task domain. It must not only know things (e.g., departure times of trains) and be able to do things (e.g., get bananas), but must know that it knows them or that it can do them. This was first recognized when John McCarthy (1968) proposed knowledge structures that included the "canult" metaknowledge structure, i.e., that the system "can ultimately" do something.

Relationship metaknowledge. Now consider the alternative, that the "indirect request" **is** actually indirect. The system parses the indirect request, "Do you know when the Windsor train leaves?" checks its metaknowledge and finds that it can indeed provide the information. It could stop there and simply say "yes." But we are assuming that this is an indirect request. How does the system know this? It needs other metaknowledge useful for inferring that "yes" is insufficient. One valuable type of metaknowledge would involve some additional knowledge about the task domain. For example, it would be useful to know that having information about train departure times is often needed to ride a train. This is information about a speaker's task goals or intentions, in this case, the goal of riding on a train.

Why did the speaker use indirection to ask when it leaves? Some theorists (e.g., Searle, 1975) say that speakers use indirection to appear polite. Now we are no longer talking about intentions related to a single task domain, such as train transportation, but to a second task domain about managing interpersonal relationships. This is communication related to a special type of metaknowledge, called "metacommunication." We finally arrive at our major theoretical hypothesis, that there are two types of metaknowledge needed to handle indirect requests: content metaknowledge about task domains, such as booking reservations, and relationship metaknowledge about managing interactions.

Behavioral Data on Use of Indirect Requests

Our research is based on a collection of 1500 requests transcribed from 27 telephone conversations between airline reservation agents and customers. In addition to our own collection of actual human behavior, some of this analysis is based on the behavioral research of others.

Frequency of use. Out of our 1500 sentences, only 17, or 1.13%, are indirect requests. This may sound surprising, considering that

interactions between airline reservation agents and customers is a context in which there are a great many requests. Although indirect requests seem to be seldom used, when they are used they should be correctly interpreted by a NLP system. Interestingly, of our 27 conversations, the 17 indirect requests appear in only 11 conversations. One conversation has three indirect requests, four conversations contain two indirect requests each, six conversations contain one indirect request each, and 16 of the conversations have no indirect requests.

Evidence on order of analysis for indirect requests. If the surface meaning is automatically interpreted first, because it would take more work and time to go on, mistakes would more likely involve saying "yes" or "no" when the requestor really wanted something more. If the surface form is ignored and the underlying meaning is processed first, then mistakes would more likely involve saying or doing more when only a yes/no response was sought. Consider if someone asks, "Do you know the time?" and receives the response, "Yes." The requestor assumes that the respondent is being uncooperative or funny; people generally do not use language in such a naïve manner. But when people give more when only a yes/no response is wanted, the requestor is more likely to accept it as a mistake in interpretation. And in reference to the time taken to process the direct vs. indirect meanings, when researchers measured the time taken by adults to interpret requests that were embedded in a story context, it actually took longer to understand surface meanings than indirect meanings (Gibbs, 1979).

Representing Metacommunication for Indirect Requests

Our representation of requests starts with a typology of 22 categories of request forms based on our research into the "imperative force" with which a request is phrased. Indirect requests express pleading imperative force and appear in six categories (see Table 1).

In addition to representing the form of requests in English, Dialogue Structures provides a representation of the conversation as a network of interacting participants (Sanford & Roach, 1988). This representation includes not only who is speaking **to** whom, but who is speaking **for** whom. That is, it is common for one person to act as a communicative "proxy" for another person, e.g., a secretary is "deputized" to speak for his/her boss and the reservation agent is "deputized" to speak for the airlines in negotiating the sale of a ticket. Usually, this "proxy/deputer" structure remains in the background. Occasionally it is made explicit such as in one of our transcripts; when an agent does not want to be blamed for the high cost of a ticket, she says, "TWA is showing me a fare, high season of $703 based out of Washington."

When two people are interacting, they usually act as proxies for separate people, that is, both speak for themselves or someone else. But occasionally, one participant speaks for the other participant in the conversation. In this case, an indirect request provides added complications. Before showing how we handle this case, let us consider the simpler case in which each is speaking for someone other that the other participant.

When an indirect request is encountered, DIALS has a set of rules for transforming such patterns into a direct, canonical form. For example, an indirect pattern such as "Do you know" is transformed into "I request to know" and "Can you help me" is transformed into "I request you help me." In the case where one speaker is acting as a proxy for the other speaker, the set of transformations are slightly different. For example, if the speaker wants to use a pay phone, "Do you have a dime?" is transformed into "I request to have a dime." But when the hearer is trying to use a pay phone and is checking his pockets or her purse, the speaker is acting as a proxy for the hearer when stating, "Do you have a dime?" and this is transformed into "I request to give you a dime."

The full set of indirect requests from our protocols and their respective direct transformations are presented in Table 1. As it shows, all 17 of the indirect requests in our protocol collection of 1500 requests are correctly transformed by DIALS. The identification of the form and imperative force of each request and the transformation to a standard form takes between 3 and 14 CPU seconds using a 1 klips PROLOG interpreter running on a VAX 11/780. DIALS can identify the category of and transform over 5000 surface forms of a single underlying request content in comparable time.

CONCLUSION

The main problem with current approaches to analyzing indirect requests is that they are trying to make inferences about the relationship component of communication by using content-based knowledge. Indirect requests are conventionalized patterns for expressing metacommunicational information. To parse them correctly requires a pattern parser that knows the forms in which requests may be embedded, how to transform such forms, and the metacommunication being expressed by the forms. There are components of the theory of Dialogue Structures that impact the analysis of indirect requests, such as elements representing the social context of conversations, that were not mentioned here due to a lack of space. Our system, DIALS, knows our behaviorally validated typology of request forms; it knows how to transform requests into and out of each category, including transforming

from indirect to direct forms; and it knows the "imperative force," i.e., the metacommunicational value, of the forms associated with each category of request pattern. This can be seen by the fact that DIALS correctly transformed all examples of indirect requests from a corpus of actual human dialogue comprised of 27 conversations with a total combined time of over 80 minutes of continuous conversation. DIALS can identify the category and imperative force of and transform over 5000 conventionalized surface forms of requests.

REFERENCES

Allen, J. (1983). Recognizing intentions from natural language utterances. In M. Brady & R. C. Berwick (Eds.), *Computational models of discourse*. Cambridge, MA: M.I.T. Press, 107-166.

Gibbs, R. W., Jr. (1979). Contextual effects in understanding indirect requests. *Discourse Processes, 2*, 1-10.

Grosz, B. J. (1980). Utterance and objective: Issues in natural language communication. *AI Magazine, 1*, 11-20.

McCarthy, J. (1968). Programs with common sense. In M. Minsky (Ed.), *Semantic Information Processing*. Cambridge, MA: M.I.T. Press, 403-418.

Sanford, D. L. & Roach, J. W. (1987a). Imperative force in request forms: The demanding vs. pleading dimension of directives. Paper presented to the International Communication Association conference, Montreal, Canada.

Sanford, D. L. & Roach, J. W. (1987b). Parsing and generating the pragmatics of natural language utterances using metacommunication. *Proceedings of the ninth annual conference of the Cognitive Science Society. Seattle, WA.* Hallsdale, NJ: Lawrence Erlbaum, 89-95.

Sanford, D. L. & Roach, J. W. (1987c). Representing and using metacommunication to control speakers' relationships in natural language dialogue. *International Journal of Man-Machine Studies, 26*, 301-319.

Sanford, D. L. & Roach, J. W. (1988). Communication and intentionality in natural language dialogue. Paper presented to the International Communication Association conference, New Orleans, LA.

Sanford, D. L. & Roach, J. W. (in press). A theory of dialogue structures to help manage human-computer interaction. *I.E.E.E. Transactions on Systems, Man, and Cybernetics.*

Searle, J. R. (1975). Indirect speech acts. In P. Cole & J. L. Morgan (Eds.), *Syntax and semantics, vol. 3: Speech acts.* New York: Academic Press, 59-82.

Table 1
18 Indirect Requests and Their Direct Transformations

Indirect Requests	CPU Secs.†	Direct Transformations
----------Modal-Auxiliaries----------		
(1) Could you make that later?	3.31	(1) I request you make that later.
(2) Can you tell me if there is a meal served on that flight?	10.14	(2) I request to know if there is a meal served on that flight.
(3) Can you tell me if it is possible to get a kosher meal on the plane?	12.92	(3) I request to know if it is possible to get a kosher meal on the plane.
(4) Well, then can you check and see if you can route us back through Pittsburgh?	13.94	(4) Well, then I request you check and see if you can route us back through Pittsburgh.
(5) Could you give me something maybe earlier than that?	5.09	(5) I request to have something maybe earlier than that.
(6) Can you tell me how much the fare will be?	5.28	(6) I request to know how much the fare will be.
(7) Could you hold on just a second?	6.25	(7) I request you hold on just a second.
----------Asking-for-Suggestion----------		
(8) Do you know how long you will be staying if you leave on the 5th?	11.75	(8) I request to know how long you will be staying if you leave on the 5th.
----------Asking-for-Permission----------		
(9) May I have your last name, sir?	4.37	(9) I request to have your last name, sir.
(10) Could I have a name and phone number?	5.55	(10) I request to have a name and phone number.
(11) Could I have your business phone number?	4.58	(11) I request to have your business phone number.
----------Asking-about-Convenience----------		
(12) Would you mind checking on that for me?	3.64	(12) I request you check on that for me.
----------Interrogative-Indicatives----------		
(13) Are there any other flights that I could take, like routed differently?	12.16	(13) I request you tell me about any other flights that I could take, like routed differently.
(14) Are there later ones?	4.00	(14) I request you tell me about later ones.
(15) Is there any later flights than that 9:55?	8.56	(15) I request you tell me about any later flights than that 9:55.
(16) Is there a morning flight?	5.02	(16) I request you tell me about a morning flight.
(17) Is there one that gets there like in the afternoon?	10.20	(17) I request you tell me about one that gets there like in the afternoon.
----------Request-to-Request----------		
‡(18) Can I ask you to check on that for me?	3.60	‡(18) I request you check on that for me.

† Measurements of time taken to perform transformations were made with a 1 klips PROLOG interpreter running on a VAX 11/780.

‡ This is an example made up to complete the categories of indirect requests. This category was not used in the 1500 requests that comprise our protocols.

Interpretation of Quantifier Scope Ambiguities

Howard S. Kurtzman
Cornell University

Maryellen C. MacDonald
Carnegie Mellon University

A sentence like (1) exhibits a quantifier scope ambiguity:

(1) Every kid climbed a tree.

Its meaning may correspond to either of the logical structures represented in (2) and (3).

(2) $(\forall x)\ (\exists y)$ (x is a kid & y is a tree & x climbed y)
(=For every kid x, there is a tree y, such that x climbed y.)

(3) $(\exists y)\ (\forall x)$ (x is a kid & y is a tree & x climbed y)
(=There is a tree y, such that for every kid x, x climbed y.)

These logical structures differ only in the relative scopes of their quantified terms. In the interpretation in (3), there is one particular tree which every kid climbed. In (2), each kid did not necessarily climb the same tree; rather, there may have been more than one tree, with different kids climbing each tree (or perhaps just one kid per tree). The first quantified term in each ordering is said to have wide scope, the second narrow scope.

Some theoretical and computational linguists have claimed, on the basis of intuitions, that the preferred interpretation of sentences like (1) corresponds to the structure in (2) (VanLehn, 1978; Kempson & Cormack, 1981; Fodor, 1982). By "preferred" it is meant that this interpretation is the one perceivers tend to arrive at first and/or to consider more likely or appropriate. Similar claims have been made for the preferred interpretation of (4) corresponding to the logical structure represented in (5) rather than in (6):

(4) A kid climbed every tree.

(5) $(\exists x)\ (\forall y)$ (x is a kid & y is a tree & x climed y)
(=There is a kid x, such that for every tree y, x climbed y.)

(6) $(\forall y)\ (\exists x)$ (x is a kid & y is a tree & x climed y)
(For every tree y, there is a kid x, such that x climbed y.)

Fodor (1982) has suggested that these preferences stem from a processing strategy according to which the preferred scope ordering of quantified noun phrases (NPs) is the same as the surface ordering of the NPs in the sentence. Such a strategy could be plausibly implemented by the construction of the initial meaning representation proceeding incrementally as the sentence is continuously processed word by word or phrase by phrase. This surface ordering strategy is essentially a claim concerning on-line processing, yet the only available data in support of this claim are off-line intuitions. There is thus no evidence concerning how strong these preferences are, whether factors other than surface

ordering influences interpretation, or whether the ultimately-preferred interpretations are the only ones available during initial processing of these sentences.

The present experiment seeks to obtain more direct evidence concerning which interpretations are available during processing of a scope ambiguity. The approach is to have subjects first read a scope ambiguous sentence and then read a sentence which is consistent with one interpretation of the ambiguity. The subjects' task is to judge whether the second sentence is compatible with the first. For example, following (1), (7) is (more) consistent with the interpretation corresponding to (2), and (8) is consistent with the interpretation corresponding to (3).

(7) The trees were full of apples.
(8) The tree was full of apples.

Similarly, following (4), (9) is consistent with the interpretation corresponding to (5), while (10) is (more) consistent with the interpretation corresponding to (6).

(9) The kid was full of energy.
(10) The kids were full of energy.

By examining subjects' responses and response times to these second sentences, it is possible to infer which interpretation(s) subjects give the first sentence. By encouraging subjects to decide as soon as they have read both sentences, data can be obtained in a more on-line task, without calling attention to the ambiguities.

METHOD

Subjects

Twenty-four college students in the Pittsburgh area were paid to participate in the experiment. All subjects were native speakers of English.

Materials

All 32 experimental passages contained 2 sentences. The first sentence was five or six words in length and contained two quantifiers. The second sentence contained six words and described one of the nouns mentioned in the first sentence.

Four variables were manipulated in the passages, each with two levels. The levels of Quantifier order in the first sentence were "every...a", as in sentence (1), and "a...every" as in (4). The first sentence Verb described either an action such as "climbed," in which case the subject of the sentence receives an agent thematic role, or a perception such as "saw", where the subject has an experiencer thematic role. The scope Interpretation variable was manipulated in the second sentence of the passage: the noun in this sentence was either singular, as in (8-9) or plural, as in (7) and (10). Finally, Ambiguity was manipulated in the first sentences by the substitution of phrases "the same" or "a different" for the quantifier "a". The unambiguous versions of (1) are illustrated in (11-12) with their

appropriate second sentence. Note that the singular/plural levels of the Interpretation variable in the second sentence dictate whether "the same" or "a different" is the disambiguating phrase in the first sentence.

(11) Every kid climbed the same tree.
The tree was full of apples.

(12) Every kid climbed a different tree.
The trees were full of apples.

The "a...every" structure illustrated in (4) was made unambiguous in similar fashion: "The same/a different kid climbed every tree".

Most of the 80 practice and filler passages contained a second sentence incompatible with the first, so that overall, the correct response on the compatibility judgment was "yes" on 60% of the trials (all experimental passages were compatible). A number of syntactic, semantic, and pragmatic violations were used to create incompatible fillers.

Procedure. Passages were presented one sentence at a time on a CRT. Subjects pressed a key once to display the first sentence and again to remove this sentence and display the second. Subjects pressed keys marked YES or NO to indicate whether the second sentence was compatible with the first. Both speed and accuracy were stressed in reading and for the compatibility judgment. Following 20 practice trials, all experimental and filler trials were presented in random order.

RESULTS

Compatibility judgments. As interpretation of the ambiguous sentences is of primary interest here, compatibility judgments for only the ambiguous sentences are reported in Table 1. Passages with unambiguous sentences were judged to be consistent 82.3% of the time, significantly more often than passages with ambiguous first sentences, min $F'(1, 44) = 23.93$, $p < .001$. This result indicates that subjects did not always have both meanings available for the ambiguous sentences.

The surface ordering strategy (Fodor, 1982) would predict that for ambiguous "every...a" sentences, the plural continuation would be preferred, while the singular continuation should be preferred for the ambiguous "a...every" sentences.[1] The data in Table 1 strongly support this prediction. For the ambiguous sentences, there was a significant Quantifier Order x Interpretation interaction, min $F'(1, 51) = 62.08$, $p < .0001$. In contrast, there was no evidence that the surface ordering of the quantified NPs affected the interpretation of the unambiguous sentences: this interaction was not reliable for unambiguous sentences, min $F'(1, 47) = 2.69$, $p > .10$. The disambiguating information in these sentences apparently overrode any surface ordering strategy.

TABLE 1

Percentage of Ambiguous Passages Judged Compatible. Experiment 1

Interpretation	"Every...a" sentences Action Verb	Perception Verb
Singular	25.0	62.5
Plural	81.3	72.9

Interpretation	"A...every" sentences Action Verb	Perception Verb
Singular	87.5	85.4
Plural	14.5	14.5

Further examination of Table 1 reveals that for the "every...a" sentences, the action/perception verb manipulation dramatically affected acceptance rates. Subjects judged the singular interpretation of these sentences to be more acceptable when the first sentence had contained a perception verb, compared to when it had contained an action verb, $\underline{\min F'}(1, 21) = 6.87$, $\underline{p} < .025$. We return to this point in the discussion.

Because the acceptance rates for the non-preferred interpretations of ambiguous sentences are so low, response times to make this compatibility judgment are difficult to interpret. However, we should note that on those trials where subjects judged the second sentence to be compatible with the first, response times for passages with an ambiguous first sentence (3752 ms) were longer than for unambiguous passages (3073 ms). This result suggests that the disambiguating information helped subjects arrive at an interpretation more rapidly. Despite the fact that these response times (which include time to read the second sentence) are rather long, this paradigm still provides a much more immediate measure of how perceivers interpret these ambiguities than can be gleaned from intuitions. Additionally, few subjects reported that they noticed any ambiguities in the stimuli, indicating that the task successfully prevented subjects from consciously searching for multiple meanings for the sentences.

DISCUSSION

This experiment demonstrated that subjects could rapidly interpret scope ambiguous sentences, and that their preferred interpretations were the ones where the first-encountered NP had wide scope over the second NP. Fodor's (1982) surface ordering proposal is clearly compatible with these data, but it is not the only interpretation available. Kempson & Cormack (1981), for example, have suggested that the preferred interpretation is the one

where the sentence topic (usually the first NP) has wide scope.

Pragmatic Influences. The data also suggest that, while the surface-based scope ordering is preferred, it is not always the only one that subjects initially assign. In particular, for the "every...a" sentences with perception verbs, subjects judged both singular and plural continuations to be compatible over 50% of the time. For this sentence type, then, it appears that subjects assigned both interpretations, although the interpretation incorporating the surface-based scope ordering (the plural continuation) was still preferred.

Why would subjects be more likely to assign both interpretations for this sentence type? First, compare the "every...a" sentences with action verbs and with perception verbs. It is quite plausible, within a single event (taken to be the referent of each of these sentences), for more than one perceiver to perceive the same entity, but it is generally somewhat less plausible for more than one agent to perform an action on the same entity. To take an extreme example not used in the experiment, it is plausible for many children to see one unique cookie, but it is pragmatically very awkward to interpret "Every child ate a cookie" as meaning that all the children jointly ate one unique cookie. Our action verb sentences (e.g. (1) and (4)) were intended to be plausible with both singular and multiple entities (e.g. trees), but the data suggest that subjects did not find them equally plausible. Thus, at least for the stimuli tested here, the interpretation in which the second NP (with the determiner "a") has wide scope is pragmatically more appropriate for sentences with perception verbs than with action verbs.

Now compare the "every...a" sentences with perception verbs with the "a...every" sentences with either type of verb. When "a...every" sentences are presented with no preceding context, as in this experiment, the subject, upon receiving the first NP (with the determiner "a"), has no reason to consider the phrase as referring to more than one entity: in fact, it would be pragmatically unparsimonious for the subject to do so (cf. Crain & Steedman, 1985). It is only when the second NP (with the determiner "every") is received that the real potential arises for the first NP to refer to multiple entities. This potential is correlated with the first NP being assigned narrow scope (i.e., with the scope ordering which does not match surface ordering). But a changed reference for the first NP, from singular to potentially multiple, is pragmatically quite significant. It is reasonable to assume that a change in the reference tends not to be determined except when all possible interpretations require it, due to the semantic processing complexities which the new reference necessitates (cf. Gillon, 1984; Crain & Steedman, 1985). Therefore, it is unlikely that the alternative interpretation for the "a...every" sentences--in which the first NP has narrow scope--is routinely assigned.

By contrast, for the "every...a" sentences, the first NP received (with the determiner "every") inherently refers to multiple entities. This is the case regardless of whether the second NP (with the determiner "a") is assigned narrow or wide scope. Because no

pragmatically significant changed reference for the first NP needs to be determined even if the alternative interpretation is assigned. multiple interpretations can be more readily assigned to the "every...a" sentences than to the "a...every" sentences.

A Processing Model. Precisely how do the pragmatic influences interact with the basic scope ordering preference? One possibility is that for all scope ambiguous sentences. both possible interpretations are determined immediately (i.e.. upon receiving the second NP. which is the point where the ambiguity arises). These interpretations vary in their levels of accessibility. based primarily upon the preference for surface-based scope ordering. Thus. following Fodor (1982). the more accessible interpretation would be the one whose scope ordering is more quickly or directly identified in continuous processing of the sentence. Or. following Kempson & Cormack (1981). it would be the one in which the sentence topic has wide scope.

The less accessible interpretation can. however. have its accessibility increased somewhat. on pragmatic grounds. Thus. if the less accessible interpretation is highly plausible and if it does not involve a changed reference for the first NP (with respect to the reference in the other. more accessible interpretation). then its accessibility can be increased. although not to the level of the other interpretation. This is the case for the "every...a" sentences with perception verbs.

Next. all interpretations whose accessibility is below a criterion level are deleted. This deletion would occur sometime between processing of the second NP and evaluation of the continuation sentence. The remaining interpretations and their accessibility levels determine how frequently subjects respond that each of the continuation sentences is reasonable. (See Kurtzman. 1985: Gorrell. 1987. for similar views of the processing of syntactic ambiguities.)

Obviously. further work is required to elaborate and experimentally test this model. Also. further work is needed to determine the nature of the mental representation of scope ordering. Among the options are: standard logical formulae (Hobbs & Shieber. 1987). indexed syntactic structure (Williams. 1988). a syntactic level of LF (May. 1985). mental and discourse-representation models (Fodor. 1982: Kamp. 1984). and propositional networks (Anderson. 1983). In a first step toward further specification of this model. we are currently conducting an experiment with passive versions of the stimuli. such as:

(13) A tree was climbed by every kid.

(14) Every tree was climbed by a kid.

Actives and passives share the same propositional meaning. but differ in the surface ordering of their NPs. Therefore. passives provide a sharp test of the claim that surface-based scope ordering is preferred. The experiment also allows further examination of pragmatic influences on interpretation. in that it provides additional opportunity for examination of the verb effects found in the present experiment. The results will be

reported in a future paper.

FOOTNOTE

1. Strictly, the interpretation in which the phrase with the determiner "a" has narrow scope is compatible with either the singular or the plural continuation sentence. That is, the logical structures in (2) and (6) specify only that it is possible--not that it is necessary--for there to be more than one tree and more than one kid. Thus, one might challenge our claim that preference for the singular interpretation for the "a...every" sentences indicates a preference for wide scope of the first NP. However, the low percentage of YES responses for the plural continuation sentences suggests that the other interpretation (narrow scope on the first NP) was not preferred.

Focussing now on the "every...a" sentences, the high percentage of YES responses to the plural continuation sentences is compatible only with a preference for narrow scope for the "a" phrase. However, as just noted, a singular reference for the "a" phrase is possible with narrow scope. Why then is the percentage of YES responses for the singular continuations lower (especially for the action verb sentences)? Apparently, when an interpretation permits either multiple or singular reference, multiple reference is preferred. Fodor (1982) has suggested how this could be accounted for if quantifiers are represented within the format of mental models. Clearly this is an area for additional investigation.

REFERENCES

Anderson, J.R. (1983). The architecture of cognition. Cambridge, MA: Harvard Univ. Press.

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. Kartunnen, & A. Zwicky (Eds.), Natural language processing. Cambridge: Cambridge Univ. Press.

Fodor, J.D. (1982). The mental representation of quantifiers. In S. Peters & E. Saarinen (Eds.), Processes, beliefs, and questions. Dordrecht: Reidel.

Gillon, B.S. (1984). The logical form of plurality and quantification in natural language. Ph.D. Dissertation, M.I.T.

Gorrell, P.G. (1987). Studies of human syntactic processing: Ranked-parallel versus serial models. Ph.D. Dissertation, University of Connecticut.

Hobbs, J.R., & Shieber, S.M. (1987). An algorithm for generating quantifier scopings. Computational Linguistics, 13, 47-63.

Kamp, H. (1984). A theory of truth and semantic representation. In J. Groenendijk, T.M.V. Janssen, & M. Stokhof (Eds.), Truth, interpretation, and information. Dordrecht: Foris.

Kempson, R.M., & Cormack, A. (1981). Ambiguity and quantification. Linguistics and Philosophy, 4, 259-309.

Kurtzman, H.S. (1985). Studies in syntactic ambiguity resolution. Ph.D. Dissertation, M.I.T. (Distributed by Indiana University Linguistics Club.)

May, R. (1985). Logical form. Cambridge, MA: MIT Press.

VanLehn, K.A. (1978). Determining the scope of English quantifiers. Artificial Intelligence Laboratory Report AI-TR-483, M.I.T.

Williams, E.S. (1988). Is LF distinct from S-Structure? A reply to May. Linguistic Inquiry, 19, 135-146.

Multiple Simultaneous Interpretations of Ambiguous Sentences

Peter Norvig
University of California, Berkeley

INTRODUCTION

This paper is concerned with the problem of semantic and pragmatic interpretation of ambiguous sentences. We start by offering a simple yet commonly adopted interpretation strategy:

> **Strategy 1:** Apply syntactic rules to the sentence to derive a set of parse trees. Next apply semantic rules to the trees to get a set of logical formulae, and discard any inconsistent formulae. Do a pragmatic interpretation of each formula, and give a score to each possibility based on consistency or likelihood in the given context. Finally, choose the interpretation with the highest score.

In this framework, the lexicon, grammar, and semantic and pragmatic interpretation rules determine a mapping between sentences and meanings. A string with exactly one interpretation is unambiguous, one with no interpretation is anomalous, and one with multiple interpretations is ambiguous. To enumerate the possible parses and logical forms of a sentence is the proper job of a linguist; to then choose from the possibilities the one "correct" or "intended" meaning of an utterance is an exercise in pragmatics or Artificial Intelligence.

One major problem with Strategy 1 is that it ignores the difference between sentences that seem truly ambiguous to the listener, and those that are only found to be ambiguous after careful analysis by the linguist. For example, each of the following is technically ambiguous (*with* could signal the instrument or accompanier case, and *port* could be a harbor or the left side of a ship), but only the third would be seen as ambiguous in a neutral context.

(1) I saw the woman with long blond hair.
(2) I drank a glass of port.
(3) I saw her duck.

Zadeh (personal communication) has suggested that ambiguity is a matter of degree. He assumes each interpretation has a likelihood score attached to it. A sentence with a large gap between the highest and second ranked interpretation has low ambiguity; one with nearly-equal ranked interpretations has high ambiguity; and in general the degree of ambiguity is inversely proportional to the sharpness of the drop-off in ranking. So, in (1) and (2) above, the degree of ambiguity is below some threshold, and thus is not noticed. In (3), on the other hand, there are two similarly ranked interpretations, and the ambiguity is perceived as such. Many researchers, from Hockett (1954) to Jackendoff (1987), have suggested that the interpretation of sentences like (3) behaves like the perception of visual illusions such as the Necker cube or the vase/faces or duck/rabbit illusion. In other words, it is possible to shift back and forth between alternate interpretations, but it is not possible to perceive both at once. This leads us to Strategy 2:

> **Strategy 2:** Do syntactic, semantic, and pragmatic interpretation as in Strategy 1. Retain only the highest-ranking interpretation(s), according to some threshold function. If there is more than one interpretation remaining, alternate between them.

A problem with Strategy 2 is that it assumes all possible interpretations will be considered and ranked. However, many sentences have a prohibitively large or infinite number of interpretations. Consider the following sentence:

(4) He seems older now.

Here *he* can refer to one of several billion males, and *now* can refer to one of an infinite number of time points. Thus, while syntax and semantics may be producing discrete lists of possibilities, it seems that pragmatic interpretation must operate by proposing likely interpretations, rather than

enumerating all possibilities and then choosing among them. Hobbs (1983) has argued that enumeration should be minimized even in syntactic and semantic analysis. Accepting that intuition, we get Strategy 3, which is similar to the approachs used by several recent authors, including Hobbs (1987), Stallard (1987), and Charniak (unpublished).

> **Strategy 3:** Do lexical, syntactic and semantic analysis to produce one (or occasionally more) neutral representation of the input, which can contain ambiguous and vague predications. Pragmatics then attempts to 'solve' for the ambiguous predications and some of the vague ones. Solutions are generated in a roughly best-first manner, and when there is a large drop-off in the ranking of solutions, we stop and the final interpretation alternates between the high ranking one(s).

This might represent an efficient interpretation mechanism, but it doesn't mirror the human interpretation mechanism particularly well. Sentences like (5-9) each have only one good pragmatic interpretation, which would be found easily by Strategy 3. But (5-9) are notoriously hard for humans to get right without at least a conscious sense of having to back up and re-parse the sentence.

(5) The horse raced past the barn fell.
(6) The astronomer married a star.
(7) The rabbi was hit on the temple.
(8) The landlord painted all the walls with cracks.
(9) Ross was told what to do by the river.

Strategies like Kimball's (1973) or Frazier and Fodor's (1978) try to account for phenomena like these in terms of general syntactic preference principles, which appeal to performance issues such as limits on available memory space. Schubert (1984, 1986) and Kurtzman (1984) argue convincingly that no simple syntactic preference will do. Rather, many factors must be considered, as in Strategy 4:

> **Strategy 4:** Do lexical, syntactic, and semantic analysis on a word-by-word basis, identifying points of ambiguity along the way, and using all sources of evidence to rank alternatives. Evidence for a particular choice can include lexical frequency preferences, pragmatic associations, and other factors outside of the simple logical form. A high-ranking interpretation can be accepted (and its alternatives discarded) before the parse is complete, if its score remains sufficiently above the alternatives for a sufficient amount of time. In addition, if at any point there are more than a maximum number n ($n = 3$?) alternatives, discard the lowest ranking alternative, even if its score is close to others. At the end, alternate between the highest ranking interpretations, as before.

MUTUALLY COMPATIBLE INTERPRETATIONS AND CONNOTATIONS

Consider the following quote from Richard Parsons, of the American Fur Industry Inc., on their new advertising slogan *Fur is for Life:*

> "It has a good sound, a good connotation. Yes, they last a long time. Yes, they're a good product. Yes, furs support wildlife conservation."

Parsons (although not a professional linguist) is making a claim about language use: that the proper or intended meaning of a phrase can be a combination of a number of interpretations and connotations. Strategies 2-4 assume that the reader eventually arrives at a single interpretation, or a Necker-cube-like alternation between interpretations. But Parsons is saying that his slogan *Fur is for Life* is different. The slogan seems to have two primary interpretations, (10) and (11)

below. But it also has important *connotations*, listed as (12-14), as well as another interpretation, (15), that Parsons presumably wants the public to ignore.

(10) Fur lasts a lifetime.
(11) The fur industry is pro-conservation.
(12) Fur wearers are lively.
(13) The recipient of a fur may become indebted to the giver for life.
(14) Life is a good thing; hence fur is a good thing.
(15) Fur, while on an animal, protects its life.

Although we would not be likely to say that any of (12-14) are good candidates for the final interpretation, it seems that the intended effect of the slogan is for the reader to entertain some or all of these simultaneously. While this is a radical departure from the Hockett Jackendoff Necker theory of 'one interpretation at a time,' it appears to be quite common in poetry, politics, and advertising (see Burli-Storz, 1980).

The facts are admittedly slippery; I am suggesting that alternative parses can sometimes be combined into one interpretation, but it is hard to distinguish between distinct parses that have been combined together, and a vague interpretation that has several possible entailments. Also, it is notoriously hard to introspect about the phenomenology of these cases. Perhaps the following example, from the last line of Gerard Manley Hopkins' *God's Grandeur* will be more compelling:

(16) Because the Holy Ghost over the bent
World broods with warm breast and with ah! bright wings.

The word *broods* is lexically ambiguous between 'to sit on eggs to hatch them' and 'to think long and deeply or resentfully.' This is clearly not a case of vagueness. Yet it seems that the most natural interpretation is of a bird-like god sitting on an egg-like world (or world-like egg), pensively surveying his creation, and waiting for it to come to fruition. This interpretation clearly involves no Necker-like alternation between senses for *broods;* rather, it involves a simultaneous synthesis of two images.

Note that not just any images can be superimposed this way; if the world were flat, or if eggs were cubical, the combined image would not work. It *is* permissible to combine the images even though the world is quite a bit larger and composed of different material than the average egg, and even though the prototypical image of God does not include wings.

Poetry, like advertising, seems to sanction this superimposition of distinct parses. To support this claim, I opened a poetry anthology at random, finding the opening line of to Dylan Thomas' poem *In the Beginning:* "In the beginning was the three-pointed star." As the rest of the poem makes clear, the three-pointed star should be taken as referring to a stellar body in primordial space, to the light in God's performative speech act "Let there be light," to the star of Bethlehem, and to the Holy Trinity. There does not seem to be a clear feeling of shifting between these referents; rather they seem to be entertained simultaneously.

Lakoff and Turner (1988) cite, but do not fully analyize, another Dylan Thomas poem, *Do not go gentle into that good night:*

(17) Do not go gentle into that good night,
Old age should burn and rave at close of day;
Rage, rage against the dying of the light.

Understanding this passage requires knowledge of at least six metaphors for life and death. While these metaphors offer conflicting views on the nature of death, there is no feeling of having to switch between them in understanding the poem; they are all active at once. In fact, metaphors (18-23) are all used in the interpretation of the six words *go gentle into that good night*: (18) for *go*,

(19) for *gentle*, (20) for *into*, (21) for *good night*, and (22) for *night*. Thus, the word *night* is being used simultaneously as a time, a destination, a container, and an adversary, all without promoting a conscious feeling of Necker-like ambiguity.

(18) Life is a journey.
(19) Life is a struggle; death is an adversary.
(20) Life is 'here'; death is a another world.
(21) Death is sleep.
(22) A lifetime is a day; death is night.
(23) Life is a fire that blazes and burns out.

At this point let us try to modify Strategy 4 to account for these new findings. There are two possibilities; we can treat the combination of two interpretations as an abnormality, and try to show how it can be sanctioned, or we can treat it as the new basic interpretation mechanism, and try to show how it can be constrained.

Strategy 5a: The Conservative Simultaneity Strategy: Ammend Strategy 5 to allow a simultaneous amalgam of two or more competing top-ranked interpretations, but only when sanctioned by some as-yet-unspecified factors, and only when the result is a coherent combination of the two.

Strategy 5b: The Radical Simultaneity Strategy: *Always* try to combine top-ranking interpretations into one image. When a coherent combination is impossible, alternate between interpretations as in Strategy 5.

To try to choose between the two, we will first consider Strategy 5b, as it is applied to sentence (24), and its interpretation, the disjunction (24'):

(24) The chicken is ready to eat.

(24') chicken(x) & ready(x,e) & eating(e) & (agent(e,x) | patient(e,x))

Using Strategy 5b, we could combine the two interpretations simply by accepting both parts of the disjunction, yielding 'the chicken is ready to eat the chicken.' This is by no means a normal interpretation of (24), so we have an argument against 6b. However, that argument only goes through if the proposed logical form (24') is accurate. Suppose we use the following logical form instead:

chicken(x) & ready(x,e) & eating(e) &
((agent(e,x) & alive(x) & location(e,barnyard) & patient(e,seed) & ...) |
(patient(e,x) & not alive(x) & location(e,table) & agent(e,human) & ...))

Then we have two interpretations that cannot be combined coherently, neither under Strategy 5b nor 5a. Thus, we see that for 5b to be feasible, we need to insist on full frame-like semantic interpretations, complete with default assumptions. We need a rich set of defaults to rule out unwanted unification of the two interpretations, even though we want to allow the possibility of overriding some of the defaults, as in "The chicken on the table is ready to eat her asparagus." Now let's try an example that does not bring as much background knowledge into play:

(25) She opened the door with a key.

The ambiguity is between *with a key* as an instrument of opening, and as a modifier of the door. Here there seems to be nothing to stop 5b from accepting both interpretations for the phrase, whereas we know that if this were the intended meaning, one would have to use something like

the following:

(26) She opened the door with the key that was in/near it.

Thus, Strategy 5b as it stands is rejected. To evaluate Strategy 5a, we need to develop a better notion of *sanctioning* a combined interpretation, which we will address in the next section.

JOKES AND PUNS

Consider the following advertisement for Flintstones brand Vitamins:

(27) We are Flintstones kids, ten million strong and growing.

The coordinate *and growing* can attach to either *are* or *ten million strong*, with the respective interpretations that the individual children are growing, or that the number of children is increasing. Most informants recognize both alternatives, but report an ability to fuse the two together into a single image where each individual child in an expanding group is growing. (However, no one interpreted *strong* as possibly modifying *kids*, perhaps because of the idiomatic nature of the phrase *ten million strong*.) My analysis of this example is that the listener arrives at the two interpretations using something like Strategy 4, and in the process of trying to choose between them, realizes that both were intended interpretations, and successfully superimposes the two images.

In short, (27) is a kind of pun. In a regular pun, the main point of the utterance is that the speaker has been clever, producing two meanings in one sentence. A secondary point is one of the meanings (and, for a good pun, both of the meanings taken separately). But in (27) we have a special kind of pun, where the point is that both meanings are to be taken simultaneously. A similar example comes from another ad, for Michelin tires:

(28) Because you've got a lot riding on your tires.

Here the ambiguous phrase *got a lot riding on* is ambiguous between 'much depends on your tires' and 'much rides in the car which is on the tires,' with the resulting combined interpretation 'your family's safety while in the car depends on the tires.' Here again the reader must recognize the 'pun,' and the intended effect of combining the two interpretations, but here there is an added hitch: it is the combination of the two interpretations that resolves the phrase *a lot* to 'your family'; neither of the two interpretations strongly point to this interpretation singly, but together they do.

Let us compare these puns to the following example from Freud (1916):

(29) I met Baron Rothschild, and he treated me quite as his equal—quite famillionairely.

This is funny, Freud claims, because of the unexpected ease of combining *familiarly* with *millionaire* to create a new word meaning 'as familiarly as is possible for a millionaire.' (In German, familiär + Millionär = familionär.) Freud also presents the standard definition of joking as the ability to find hidden similarities between dissimilar things. This is amended to allow for the discovery of differences, or just "to bind into a unity, with surprising rapidity, several ideas which are in fact alien to one another." In other words, the combination of disparate ambiguous interpretations is an unusual event, but one that we have an automatic capacity for.

A remaining problem is to explain why some such ambiguities are funny, while others are not. Why is it that, to my ears at least, *the rabbi was hit on the temple* is funny, while *the plumber lit his pipe* is merely confusing? Freud claims that the laughter response is illicited by the release of suppressed violent or sexual thoughts. That explains, perhaps, why the following is a fairly good

joke, while other lexical and structural ambiguities in this paper are not:

(30) She criticized his apartment, so he knocked her flat.

Minsky (1980) recasts Freud's notions into the terminology of mental agents acting as censors to violent or sexual thoughts. In Minsky's terms, certain mental agents are good at combining ambiguous interpretations, but other agents notice that this is not the normal mode of operation, and act to censor them. The laughter response serves to 'shake up' the mind, get it back on track, and post a warning to avoid such thoughts. Presumably, the simultaneous combinations that sneak by uncensored are ones that do not represent 'dangerous' modes of thought.

SIMULTANEOUS INTERPRETATION IN 'NORMAL' LANGUAGE

There are also cases of combined simultaneous interpretation which don't involve poetic license or puns. Consider the use of *book* in (31). *Book* is polysemous between a physical object, a string of words, and an abstract plot or sequence of situations. The use of *beautifully bound* refers to the physical object, *one new idea* refers to the abstract content, and *50,000 words* refers to a particular (abstract) instantiation of the content. (If the book were reprinted in paperback it would still have the same number of words, whereas if it were translated into another language, it would have a different number of words, but the same number of ideas.) All three polysemous interpretations of *book* are used simultaneously.

(31) This book, although beautifully bound, contains only one new idea in 50,000 words.
(32) He is the author of over 100 books.

It can't be that *book* is a single sense implying all these aspects, because in (32), *book* must refer only to the 'plot or sequence of ideas' sense. One could not felicitously use (32) to describe someone who had written a single book which has had a hundred copies printed, or a single book which was translated by others into a hundred languages.

Len Talmy (1977) provides a good example of image combination in non-ambiguous language. In (33), the single interpretation is 'she traveled lightly and easily through the room and the guests at the party, her path displaying a topology similar to a leaf wafting through air.' (33) forces the reader to combine the image of a woman walking through a party with the image of a leaf wafting through the air (or something similar) to arrive at the result. Talmy explains just what properties of the verb are maintained, and which are taken from the complements.

(33) She wafted through the party.

Image combination is more obvious in the case of metaphors and cliches where the derived meaning is removed than the surface form. Compare (34), which is a consistent use of metaphor, with (35). Sentence (35) provides a topological clash that cannot easily be resolved into a single interpretation, even though the meaning of the two cliches is consistent.

(34) I've always been 100% behind my husband, pushing him on as best I can.
(35) I've always been at my husband's side, 100% behind him.
(36) They can't afford to get out from under the rat-hole of rent payments.

Sentence (36), taken from a newspaper article on real estate prices, is an example of a mixed metaphor with varying effectiveness; some find it to be fine, while most report that the topology is all wrong: one should be striving to get out of a rat-hole, not out from under it.

CONCLUSION

In this paper I have investigated several strategies for pragmatic interpretation, and have presented a new strategy which (1) accounts for the little-mentioned phenomenon of a simultaneous combination of ambiguous interpretations, (2) is not inconsistent with experimentally derived human preference results, and (3) uses a combination mechanism that is needed for non-ambiguous language as well.

ACKNOWLEDGMENT

The author has been sponsored by the Defense Advanced Research Projects Agency (DoD), Arpa Order No. 4871, monitored by Space and Naval Warfare Systems Command under Contract N00039-84-C-0089.

BIBLIOGRAPHY

Burli-Storz, Claudia, *Deliberate Ambiguity in Advertising*, Bern, 1980.

Frazier, Lyn and Janet Dean Fodor, "The sausage machine: A new two-stage parsing model," *Cognition*, **6**(4), December 1978, 291-325.

Freud, Sigmund, *Jokes and Their Relation to the Unconscious*, Translated by James Strachey, 1960, Norton, NY, originally published 1916.

Hobbs, Jerry, "Representing ambiguity," *Proceedings of the Second West Coast Conference on Formal Linguitics*, Stanford University, 1983.

Hobbs, Jerry, and Paul Martin, "Local pragmatics," *Proceedings of the International Joint Conference on Artificial Intelligence*, 1987.

Hockett, Charles F., "Two models of grammatical description," Word, 1954, 386-399.

Jackendoff, Ray, *Consiousness and the Computational Mind*, MIT Press, 1987.

Kimball, John, "Seven principles of surface parsing in natural language," *Cognition*, **2**(1), 1973, 15-47.

Kurtzman, Howard S., "Ambiguity resolution in the human syntactic parser: An experimental study," *Coling*, 1984.

Lakoff, George, and Mark Turner, *More Than Cool Reason*, to appear, 1988.

Minsky, Marvin, "Jokes and the logic of the cognitive unconscious," *AI memo No. 603*, MIT AI Lab, 1980.

Schubert, Lenhart K., "On parsing preferences," *Proceedings of Coling84*, Stanford, CA, July 1984.

Schubert, Lenhart K., "Are there preference trade-offs in Attachment Decisions?," Proceedings of AAAI, 1986, 601-605.

Stallard, David, "The logical analysis of lexical ambiguity," 25th Annual Meeting of the ACL, 1987, 179-185.

Talmy, Leonard, "Rubber-sheet cognition in language," *Chicago Linguistics Society*, **13**, 1977, 612-628.

The Role of Analogy in a Theory of Problem-Solving

Beth Adelson, Mark Burstein, Dedre Gentner, Kristian Hammond, Keith Holyoak, Paul Thagard

The processes that underlie the generation and use of analogies have consistently been of interest in the study of cognition. The goal of this symposium is to look at several computational theories of analogy and to see in what ways each contributes to our understanding. We hope to attain this goal by looking at the sufficiency of each one in the broader context of problem-solving and by asking the following questions:

1. In what way is a theory of analogy constrained by specifying its role in a theory of problem-solving?
2. To what extent are the theories presented below competing theories? To what extent are they members of the same class? Does each one speak to any issue that the others do not address?
3. How is structure related to purpose and semantics?
4. How do purpose and semantics affect:
   (a) Retreival
   (b) Mapping
   (c) Justification
   (d) Debugging
   (e) Generalization

   Present an example reflecting your theory's position.

The Structure-mapping Engine: A Cognitive Simulation of Analogy

Dedre Gentner, University of Illinois, Champagne-Urbana[1]

Computational modeling of analogy.

The Structure-mapping Engine (SME), written by Brian Falkenhainer and Ken Forbus, is a computer simulation of Gentner's structure-mapping theory of analogy (Falkenhainer, Forbus, & Gentner, 1986, in press; Gentner, 1980, 1983, 1988). Given predicate calculus representations of two potential analogs, it uses purely structural principles – *one-to-one correspondence*, *structural consistency*, and *systematicity* – to interpret and evaluate an analogy between two situations. It operates by first finding all possible relational identities between base and target; it then assigns each of these match hypotheses an evaluation, based on the structural closeness of the match and on a kind of local systematicity by which a given pair of matching predicates is assigned a higher evaluation if their parents also match. SME then sweeps these matching pairs into the largest possible sets consistent with the structural constraints laid out above and computes an overall evaluation. In addition, it hypothesizes *candidate inferences*: new facts about the target domain that are derived by analogy with the base domain. Thus, SME simulates both the *matching* of existing predicates in the two domains and the *carryover* of hypothesized predicates from one domain to the other.

There are some other points to note about the simulation:

1. SME's evaluation concerns only structural soundness. The *validity* of the inferences in the target and the contextual *relevance* of the inferences must be evaluated by separate processes (See Burstein, 1983; Collins & Burstein, in press; Gentner, 1988).

2. To allow us to check our modeling assumptions individually, SME is constructed modularly. For example, different kinds of structural evaluation rules can be tested, and (as discussed below) different predicate-matching rules can be utilized.

[1] Researchers who contributed to this work include Ken Forbus, Brian Falkenhainer, Mary Jo Rattermann, Bob Schumacher, and Janice Skorstad. This research is supported by the Office of Naval Research, Contract No. N00014-85-K-0559.

3. To my knowledge, SME has the greatest range of application of any existing analogy program. Over 40 analogies have been run successfully – that is, they have yielded human-like interpretations and evaluations. Further, SME is efficient. It takes only seconds for most examples.

4. In addition to simulating analogy, SME can also be used to simulate other kinds of similarity: e.g., *mere-appearance* matches, in which only low-order information such as object attributes are considered, and *literal similarity* matches, in which both relational structure and object properties are considered. This enables us to simulate different aspects of human similarity processing.

Psychological studies of access and inference.
Besides analogical mapping, there are other subprocesses in analogical reasoning. Given a current problem (the *target situation*), the person must *access* a similar base situation, *create a mapping* from the base to the target, *draw new inferences* on the basis of the mapping, and *judge* the soundness of the analogy and the relevance and target validity of the candidate inferences. In our recent research we examine the determinants of these subprocesses, using SME to computationally model the results of psychological experiments.

In a series of studies, we gave people different kinds of similarity matches to discover (a) which kinds of matches lead to reminding and (b) which kinds of matches are considered inferentially sound (Gentner & Landers, 1985; Rattermann & Gentner, 1987). Subjects were given roughly 30 short stories to read and remember. A week later, they returned and read a new set of stories; they were to write down any of the original stories that they were reminded of while reading the new stories. The new stories were designed to match the original stories, either as structural analogies or as superficial mere-appearance matches. The results show a dissociation. In rating soundness, subjects rated on the basis of relational commonalities: analogies were rated high and mere-appearance matches low. But their natural remindings showed the opposite pattern: superficial matches were far more likely to be retrieved than relational matches (Holyoak & Koh, 1987; Ross, B. H., 1984). Thus the matches that came most readily to memory were not the matches subjects found inferentially sound.

We have compared the performance of SME with that of our subjects for a subset of the stories (Skorstad, Falkenhainer & Gentner, 1987). We find that the results of the soundness task are best fit by running SME in analogy mode, while the results of the access task are best fit by running SME as a mere-appearance matcher. These results suggest that surface similarity is important in determining access to similarity matches, while relational similarity is important in judging the soundness of a match.

Analogical Problem Solving: A Constraint Satisfaction Approach

Paul Thagard, Cognitive Science Laboratory, Princeton University
Keith Holyoak, Psychology Department, UCLA[2]

We are developing a general cognitive architecture for problem solving and learning in which analogical problem solving will have an important role. We are aiming for a system that will incorporate all the standard components of analogical problem solving in which a source problem is used to help solve a target problem: (1) the retrieval of a potentially useful source, (2) mapping of the target to the source, (3) transfer of the solution of the source to provide a solution to the target, and (4) learning that will facilitate later problem solving, for example by forming schemas that abstract from the source and target.

Our first implementation of analogical problem solving in the PI[3] system was found to put insufficient constraints on the mapping and retrieval processes (Holland, Holyoak, Nisbett, & Thagard, 1986; Holyoak & Thagard, 1986). Accordingly, we have been developing new theories of mapping and retrieval that specify a collection of principled constraints on when an analog will be retrieved and on how components of two analogs will be mapped to each other. Mapping and retrieval both involve structural, semantic, and pragmatic (purpose-directed (Burstein & Adelson, 1987, 1988)) constraints, although the constraints vary in importance to the two processes, with semantics being more important for retrieval than for mapping.

[2]This research is supported by Contract MDA903-86-K-0297 from the Army Research Institute.
[3]Processes of Induction

In mapping, the central constraint is structural correspondence between the two analogs, a relation whose importance has been emphasized by Dedre Gentner. We maintain, however, that semantic correspondences relating predicates with similar meanings are also important. Moreover, pragmatic factors involving the purpose of the analogy can also play a role in mapping. For example, if the purpose of the analogy is to convince someone of a conclusion, then mappings that support this conclusion will be encouraged.

Connectionist models provide a graceful means for *simultaneously* satisfying multiple constraints. Accordingly, we have implemented our theory of mapping in a program called ACME[4] that takes two analogs as inputs and constructs a network of hypotheses concerning what components of the two analogs to map to each other. ACME has now been applied to more than 20 complex analogies drawn from several domains, including radiation problems of the sort investigated experimentally by Holyoak.

Complementary to ACME, we are now developing ARCS [5], a constraint satisfaction model of *retrieval* that uses semantic, structural and pragmatic constraints to help find relevant analogs stored in memory (Holyoak & Thagard, 1987). In contrast to ACME, semantic constraints take precedence in ARCS, with the retrieval of analogs initiated through associations of semantically similar concepts. However, the retrieval process is also guided by structural correspondences and pragmatic import. ARCS is also a connectionist program and is being tested on a large data base.

Eventually, we plan to integrate ARCS and ACME with our rule-based problem solver PI, producing an architecture capable of both analogical and non-analogical problem solving.

Purpose Guided Analogical Learning and Reasoning

Beth Adelson Tufts University, Cambridge, MA[6]

The goal of this research program, conducted jointly with Mark Burstein, is the development of a theory of *purpose-guided* analogical learning and reasoning. Our current work focusses explicitly on the role of partial models in the generation of analogical mappings, and suggests that the process of integrating these multiple analogies which form partial explanations can be described using a set of general principles for relating partial mental models of different types (Burstein & Adelson, 1987).

Our approach of specifying mapping using a principled set of partial models is based on the fact that one typically knows a large amount about a familiar domain and what is mapped from the familiar domain to the domain being learned, is constrained by the *purpose* of the analogy(Burstein & Adelson, 1988; Holyoak & Thagard, 1986). Additionally, although behavior, mechanism, or physical and functional topology may be focussed on during initial learning, full understanding of a complex domain requires the integration of these aspects. In what follows we present some of the issues generated by our theory.

Purpose guided debugging: The role of within domain analogy

Because analogies, by definition, do not provide perfect models of a target domain, a newly mapped model will need to be debugged; structure will need to be refined, added or dropped. Here as in other aspects of analogical learning, purpose can help to constrain the process.

For example, in one of our protocol experiments a student was taught about the concept of stacks by analogy to a stack of plates in a cafeteria. The student was then asked to write the Pascal procedure for pushing elements onto the stack. Using the plate analogy, the student was able to draw correctly a box and arrow representation of the steps involved in push. However, the student was not able to then write the first line of code in which the new element is placed on the stack by pointing the new element's next-pointer to the element that previously was the first element. What needs to be done here is to *assign* to the new element's next-pointer variable the value contained in the head-pointer variable (NEW:NEXT := HEAD). When the student was reminded that pointer variables were analogous to other types of Pascal variables he was able to write the code. That is, he used this within domain analogy to refine his representation of pointer variables.

[4] Analogical Constraint Mapping Engine

[5] Analogical Retrieval by Constraint Satisfaction

[6] This research is supported by the National Science Foundations Engineering Design and knowledge and Data Base Programs

They then took on the properties of other types of Pascal variables; they contained values of a specified type and these values could be copied using the assignment operator.

As this example illustrates, when the learner's task is to describe how a system will be realized in the target domain, base domain analogies will most likely be insufficient. However, within domain analogies may the type of analogies that are appropriate both for this purpose.

Integrating behavioral, causal and topological models

Our theory also addresses the nature of the relationship between behavioral, causal and topological models, how this information is used and why it is important in any problem-solving in which an old solution will be transformed to solve a new, similar problem.

Frequently problem solving will involve understanding the behavior of a system in terms of the relationship between input and output or start and goal states. This description of the system's goal that is contained in a *behavioral* model provides an explanation of the system's purpose. By comparison, a *causal* model represents a system as a connected set of components with causal effects and constraints. The causal model consists of a description of how the outputs of components cause state changes in other, topologically connected components. As a result, a causal description provides an account of what happens across the system in order to get from a start to a goal state. A *topological* model is needed to describe the physical realization of a system's mechanism and behavior. A topological model does this by describing both the *functionality* of the system's components and the interconnections among them.

In our theory, the three models are related in the following way. The behavior model describes the purpose for which the system is used. The causal model is related to the behavioral model in that it describes what is done to acheive the purpose stated in the behavioral model. The topological model explains how the causal model is realized.

The need for the three models and for rules describing their relationship is illustrated by the same student learning about the concept of a queue by analogy to the concept of a stack. The student was told that a queue is like a stack except that it is used when First In First Out behavior is desired. The student was then asked to write the Pascal procedures for pushing and popping stack elements.

The student knew that the Last In First Out behavior of a stack was obtained through a mechanism in which the next element to be popped was the one that was *most* recently pushed. Using the difference in the *behavior* of the two concepts he mapped a transformed *causal* model of a queue, in which FIFO behavior would be obtained by popping the *least* recently pushed element. On the basis of the newly mapped causal model the student was then able to map a transformed *topological* model of a queue. In this model, pushing and popping occur at the same, rather than opposite, ends. Also, as a result of understanding the way in which causal and topological models are related, the student was able to state that the topological model for a queue, as opposed to a stack, contains two pointers rather than one, in order to indicate separately where the next push and pop occur.

Here, and generally in situations in which existing mechanisms are used as analogies in order to implement some new desired functionality, a description of the relationship between the behavioral, the causal and the topological representation enters strongly into successful problem-solving.

Analogical Explanations Combining Inconsistent Mental Models

Mark Burstein BBN Laboratories, Cambridge, MA[7]

Analogies are used during problem solving for a number of purposes. They can be used to suggest plans of action, organizations for components in a synthesis or design problem, predictions of effects of partially understood systems, and explanations of observed behaviors. All of these uses of analogy share some common underlying cognitive mechanisms, but emphasize and exploit semantically different relational structures. We reported a preliminary categorization of some of these types of relational structure in (Burstein & Adelson,

[7] This research was supported in part by the Army Research Institute.

1987).

As described in the preceding section, we are developing a theory of analogical reasoning and learning that differentiates these different uses of analogies. One important aspect of our theory is that analogies can compare and relate systems at several levels of description. Well understood domains or systems, i.e., those whose behavior, mechanism, and applications are all understood, may serve as source analogs at several different levels of abstraction, *and* for several of these different purposes.

One recent computer model that demonstrates some of these uses of analogy is Falkenhainer's PHINEAS system (Falkenhainer, 1987). PHINEAS is an ambitious system that coordinates some large programs for qualitative reasoning and planning, in conjunction with his Structure Mapping Engine (SME) (Falkenhainer, Forbus & Gentner, 1986). Falkenhainer's system first (1) builds a qualitative causal model of heat flow by analogy to liquid flow to *explain an observation* of heat flow *behavior*. The system then seeks to (2) *verify further predictions* from its new model, as a means of testing its reliability and generality. This step also involves (3) *planning experiments* to bring about other situations predictable from the same qualitative model.

One of the reasons the behavior of PHINEAS differs from the model we are developing is that PHINEAS does not use analogical reasoning from its source domain knowledge in steps 2 and 3. That is, its predictions of related target domain behaviors and it's plans for actions to bring about those behaviors are generated using only the newly constructed target domain qualitative model. PHINEAS does not have access to stored plans suggesting uses of water flow for different purposes, nor does it use alternate envisionments of water flow situations to generate additional heat flow predictions. Its resulting model of heat flow is embeded in a specific setting based on the initially presented situation. We would argue that creation of a full test of the generality of the newly derived heat flow *principle*, and the analogy to fluid flow, would involve *extending* the analogy to relate a number of "parallel" water and heat flow situations (Burstein, 1988).

The point of this example is that successful, strategic use of analogies in learning and problem solving is often based on several levels of correspondence between domains. PHINEAS and other systems have successfully shown how the discovery of corresponding *behaviors* can lead to the induction of a corresponding qualitative *causal structure*, but many analogies can be used to relate other levels of description as well. We are developing a computer model where several levels of description can be successively mapped from a single analogy, based on an initial, successful mapping at the behavioral or some other level. This requires richer source domain representations with multiple related examples and mechanisms which allow semantically different kinds of structural relations to be preferred for mapping at different times.

Another reason for investigating this form of purpose-constrained analogical mapping is that human mental models of domains are are seldom complete, nor totally internally consistent (Collins & Gentner, 1987; Spiro et al., 1988; Burstein, 1985). We often use a *set* of related examples with associated explanatory models to keep from generating erroneous predictions and explanations. Similarly, explanations of different behaviors within a single domain may be generated from different analogical models, or combinations of models. Inconsistencies in these explanations can only be detected during problem solving episodes that fully exercise the interrelations between models. We will present examples of subjects developing and relating models derived from different analogies, and show how detected inconsistencies can be understood in terms of attempts to relate mental models at different levels of abstraction during problem solving.

Analogical Reasoning as a By-Product of Problem-Solving.

Kristian J. Hammond, University of Chicago[8]

We *don't* work on analogy. We *do* work on problem solving. As it turns out, we have a program and an approach that results in "analogical" behavior. Simply stated, the program (POLYA) seeks cases in memory to use as exemplars and often finds cases that an observer would see as analogical. POLYA itself,

[8] This work was done with Thomas McDougal at the University of Chicago AI Lab.

however, does not distinguish between true analogy and similarity. POLYA simply uses mapping information associated with the shared predicates to transfer the solution or partial solution of the recalled problem to the new situation without distinguishing between types of similarity.

POLYA is a problem-solver in the domain of geometry theorem proving. In particular, it is a *case-based* problem-solver that constructs new proofs out of existing solutions (or generalizations of solutions) that it finds in memory. The aim of the POLYA project is the development of techniques for automatically constructing vocabularies of feature combination and interaction that are effective in describing problem situations and thus organizing solutions in memory.

The POLYA project is predicated on the notion that for every domain, there exists a vocabulary of feature combination and interaction that best describes the problems within it. The two features that are most important to this vocabulary are the near independence of the predicates and a close association between problem descriptions and solution sets. The near independence of predicates enables use in indexing. The association between problem description and solution set determines utility.

This vocabulary is generated out of the constraints of existing problem/solution pairs and is tested through its use in indexing cases in memory. Those statements used in a proof that, by necessity, share arguments are compounded into a single predicate. A special purpose mechanism is then constructed to recognize this predicate and the predicate is added to the list of features used to index the solution in memory. As new problems are presented, they are understood in terms of the new predicates created by the system and these descriptions are used to search for existing solutions. Rather than confront the general problem of indexing off of arbitrary conjuncts of predicate calculus statements (potentially exponential), only certain conjuncts are recognized and used.

While POLYA constructs these new predicates they do not differ formally from those it initially uses. While a compound predicate such as SQUARE-WITHIN-CIRCLE or KITE-WITH-CROSS represents interactions between parts, the low-level predicates such as LINE and CURVE do as well. Further, the techniques for recognizing the later are of the same type as those for recognizing the former. Rather than look for analogies, POLYA simply looks for useful exemplars.

There are five steps to the POLYA architecture:

1. Proofs are used to generate compound predicates out of conjuncts of predicates implicated in the proof that share variables.
2. New problems are analyzed in terms of these predicates.
3. The resulting predicate list is used to search for appropriate solutions.
4. Mapping information associated with each predicate is used to obtain the bindings between source and target.
5. Any gaps in the proof are dealt with as a new problem description and POLYA recurs on the search for a solution.
6. The resulting proof is used to generate a new set of candidate predicates.

The result of this is a system that makes use of cases that are both "similar" and "analogous" to its problem situation without having to distinguish between the two. The behavior is analogical, but this is a natural by-product of the case-based problem solving.

References

Burstein, M. H. (1983). A model of learning by analogical reasoning and debugging. In *Proceedings of the International Conference on Artificial Intelligence*, Washington, D. C.

Burstein, Mark H. (1985). Learning by Reasoning from Multiple Analogies. Doctoral dissertation, Yale University.

Burstein, Mark H. (1988) Incremental Learning from Multiple Analogies. *ANALOGICA: The first workshop on analogical reasoning*. Los Altos, CA: Morgan Kaufmann.

Burstein, M. H. and Adelson, B. (1987) Analogical Learning: Mapping and Integrating Partial Mental Models. In *Proceedings of the 1987 Conference of the Cognitive Science Society*. University of Washington, Seattle, WA.

Burstein, M. and Adelson, B. (1988) Analogical Reasoning for Learning. in *Applications of Artificial Intelligence to Educational Testing*. R. Freedle (Ed.) In press. Erlbaum: Hillsdale, NJ.

Collins, Allan and Gentner, Dedre. (1987) How People Construct Mental Models. In N. Quinn and D. Holland (Eds.), *Cultural Models in Thought and Language*. Cambridge, UK: Cambridge University Press.

Collins, A., & Burstein, M. (in press). A framework for a theory of mapping. To appear in S. Vosniadou and A. Ortony (Eds.)*Similarity and analogical reasoning*.

Falkenhainer, B., Forbus, K. D. & Gentner, D. (1986). *The structure-mapping engine Proceedings of the Meeting of the American Association for Artificial Intelligence*. Philadelphia. Revised version to appear in *Artificial Intelligence*, in press.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7(2), 155-170.

Gentner, D. (1988). Analogical inference and analogical access. In A. Prieditis (Ed.), *Analogica*. Los Altos, CA: Morgan Kaufmann.

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* (pp.607-613). Tucson, AZ.

Holland, J., Holyoak, K., Nisbett, R., and Thagard, P. (1986). Induction: Processes of inference, learning, and discovery. Cambridge, MA: Bradford Books/MIT Press.

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

Holyoak, K. and Thagard, P. (1986). A computational model of analogical problem solving. To appear in S. Vosniadou and A. Ortony (eds.), Analogy, Similarity, and Thought. (Cambridge: Cambridge University Press, in press.)

Holyoak,K. and Thagard, P. (1987). Analogical mapping by constraint satisfaction: A computational theory. Unpublished manuscript.

Ross, B. H. (1984). Remindings and their effects in learning a cognitive skill. *Cognitive Psychology, 16*, 371-416.

Skorstad, J., Falkenhainer, B., & Gentner, D. (1987, August). Analogical processing: A simulation and empirical corroboration. In *Proceedings of Meeting of the American Association for Artificial Intelligence*, Seattle, WA.

Falkenhainer, Brian. (1987) Scientific Theory Formation Through Analogical Inference. In *Proceedings of the Fourth International Workshop on Machine Learning*. Los Altos, CA: Morgan Kaufmann Publishers, Inc.

Spiro, R.J., Feltovich, P.J., Coulson, R.L., Anderson, D. (In Press) Multiple Analogies for Complex Concepts: Antidotes for Analogy-Induced Misconception in Advanced Knowledge Acquisition. In S. Vosniadou and A. Ortony (Eds.), *Similarity and Analogical Reasoning*. New York, NY: Cambridge University Press.

THE ARCHITECTURE OF CHILDREN'S PHYSICS KNOWLEDGE A PROBLEM-SOLVING PERSPECTIVE

Kathleen E. Metz
School of Education
University of California, Berkeley

ABSTRACT

The project investigated the nature of young children's physics knowledge and the architecture of its development. I utilized two contexts of development for this purpose: comparison of cross-age developments in knowledge of a domain and fine-grained analysis of developments that occurred in the context of problem-resolution. The empirical base consisted of three conditions, under which preschoolers were asked to establish equilibrium on the pan balance. Analysis focused on the child's transformation of a number-based to a weight-based approach to the problem. All the conditions employed the same nine sets of elements to be balanced; the conditions varied *a*) whether or not the child received feedback from the apparatus and *b*) order of set presentation (total n= 56). A sequence of fine-grained analyses of the videotaped data lead to a view of children's physics knowledge as localized and context-sensitive; with the steps involved in its development as remarkably limited in extension : in *a*) the scope within which they come to represent weight or weight differences (e.g. discrete elements versus collective weight of elements in a pan) *b*) the scope ot contexts in which they come to view weight as relevant to the goal of mechanical equilibrium and *c*) the bounds of diagnostic and causal implications.

INTRODUCTION

The project investigated the nature of young children's physics knowledge and the architecture of its development. More specifically, what is the nature of the knowledge that the child brings to bear in the resolution of physics tasks? How does one approach to the task become transformed into another; i.e. in what form(s) does change enter in and how does the new become elaborated into a another more adequate approach to the task?

The literature posits incompatible models concerning the architecture of children's developing scientific knowledge. On the one hand, one finds models emphasizing the impact of improvement in *general* structures, processing capacities, or theoretical constructs, viewed as leading to broad-based improvements in the child's scientific knowledge. In Inhelder and Piaget's (1958) model, development of the underlying logical-mathematical structures enables a more adequate experimental plan and more adequate interpretation of empirical results. In a more physics-framed model, Kaiser, McCloskey and Profitt (1986) look to the adequacy of the general theoretical constructs held by the child, which they argue largely account for the nature of the child's performance in a broad range of relevant tasks. On the other hand, one finds models such as DiSessa's (1987) and Lawler's (1985) that characterize knowledge and its evolution as *localized and fragmented* and take a more particulate view of what a child knows and the steps involved in knowledge-transformation.

This project exploits both the *macro*-developmental context of cross-age improvements, as well as the *micro*-developmental context of task-based learning, to examine the architecture of children's evolving knowledge of a

physics domain. Inhelder and her research team have argued that micro-analysis within the problem-solving context is much better suited to elucidate how knowledge-structures become transformed than comparison of cross-age differences employed by most studies (cf. Inhelder et al, 1976; Karmiloff-Smith & Inhelder, 1974). This tact enables fine-grained analysis of the nature of the chunks and pathways of connections formed in the transformation of children's domain-specific physics knowledge.

RESEARCH BASE

Subjects

The projects drew subjects from a university laboratory school, of upper-middle and middle S.E.S.. Each of the first two conditions included 24 preschoolers, evenly divided across sexes and among 3, 4, & 5 year olds. The last condition included 8 five year olds, evenly divided across sexes.

Experimental Set-up

Materials consisted of a pan balance and 9 sets of little people. The standard figure was plasticine throughout, weighed one ounce, and was 5 *cm.* in length. Other figures varied in size and/or weight from the standard, by means of hidden styrofoam or hidden weights or by simple addition of more plasticine. The number of figures in a given set ranged from 2 to 9.

The child sat at a small table, facing the V.C.R. camera. The apparatus rested on the table, directly in front of the child. E sat to S's right, from which she handed S the sequence of people sets.

Experimental Procedures

In each condition, E began by instructing S, *"The people go in these things* [the pans]. *When all the people are in these things, it should stay like this"* [horizontal position of beam demonstrated]... As S engaged in the process of placements and immediately thereafter, E probes " *What are you doing?*' or *What did you do*?". If S fails on the first attempt, E encourages repairs by saying "*Try to fix it.*". E presents the next set of elements after S has succeeded with this first set (of two identical elements).

Under the first two conditions, E continues in this same mode with another eight sets of figures. Subjects receive feedback from the apparatus in each of these conditions; the conditions vary the order of set presentation. In the first order, the first few sets can be successfully placed via an equal number strategy. The other order introduces goal-relevant inter-object differences earlier on. In the third condition, following successful placement of the first set, E tells S, *"Now the game is to put the people <u>under</u> the pan they should go into. You don't put these things in anymore. You put the people under the pan they belong in".* This no-feedback condition employed the first order of the feedback condition.

FRAMEWORKS OF ANALYSIS

The author and her research assistant carried out an extensive series of analyses of the videotapes. This paper considers two: analyses of strategies employed and the much finer-grained analysis of extent of knowledge-strand elaboration. In both cases, the two independently coded the complete data set. Differences were resolved by means of joint reviewing of the tape.

Analysis One: Strategies employed

This top-level analysis consisted of coding the strategies each child employed, to obtain the goal-state of mechanical equilibrium. We parsed as an episode each attempt to get the apparatus to work, either initial placement strategy or repair.

The schema of strategies, formulated across the course of two pilots, included the following categories:

equate # in two pans
one-to-one correspondence of placements
equal weight in two pans
visual feedback
compensation by force of hands (yank on wires)
correct spatial alignment (gently place apparatus into position)
fiddle with knobs (knobs built into the beam)
reject (element or pair of elements from the current set)
exchange (pair or complete sets of elements across pans)

Each coder assigned one or more strategies to each attempt at the goal. Where more than one strategy was assigned, the coder specified the form of their combination: whether the child's actions integrated the two strategies (as in the application of one-to-one correspondence, stopping after each pair of placements to check visual feedback; and where one pan rests higher, placing just one there instead of an element in each) or where the child employs one strategy as a means to another (e.g. equating weight, by means of equating number under conditions of identity of elements in the set). After completing the assignment of strategy-codings, we reviewed the protocols for stable patterns of strategy application. These patterns encompassed the distinctions that the child used in determining initial placements and the basis on which he or she selected repairs.

Analysis Two: Extent of knowledge-strand elaboration

This finer-grained analysis aimed to elucidate how one stable pattern of strategy- application became transformed into a more adequate approach to the problem. Given the fine-grain employed here, we restricted this analysis to the 24 Study I subjects. The analysis focused on the child's knowledge of two aspects of the task domain *vis-a-vis* this goal of mechanical equilibrium: *a)* knowledge of the pan balance apparatus and *b*) knowledge concerning the sets of elements to be balanced; as well as *c*) knowledge of connections between the two.

Each coder specified: 1) the set of features by which S had represented each aspect of the domain 2) which of these features he or she conceptualized as relevant to goal-attainment and 3) the entailments of this feature for goal-

attainment. We specified the entailments S had elaborated in terms of diagnostic and causal implications.

We define diagnostic and causal implications as follows. The child who constructs a link of the type *when x happens that means y is the problem* has formed a diagnostic implication (e.g. when one pan goes down, there are too many things in it.) The child who constructs a link of the type *when you do x, y happens*, has formed a causal implication (when you put the same number in each, it works).

The framework underlying this last analysis thus distinguishes between representing some variable within some aspect of the task domain from the recognition of that variable as relevant to task-attainment. The aspect may actually be relevant to goal-attainment (e.g. relative weight of the elements) or irrelevant (color of the elements) or imperfectly correlated with the relevant (size of the elements).

By specifying the set of implications the child has constructed involving the variable and goal-attainment, the analysis can capture degrees of feature-goal elaboration: from the dawning of relevancy prior to specific implications (a suspicious "*Why did you give me a heavy one*?") to the construction of all the entailments of the feature for goal-attainment.

RESULTS AND DISCUSSION

Patterns of strategies employed

Across the 1093 data-points of strategies employed, a 91.3% inter-rater reliability was achieved. Two patterns of strategy application and a third intermediary, more transient approach, appeared across the data set. Using a criterion of 80% fit of strategies employed, application of these patterns correlated with age. Furthermore, we found an invariant order of the three, within protocols that exhibited learning.

In the first approach, the children began by placing the same number of elements in each pan (by means of one-to-one correspondence or equal number). When this failed, they would try to fix the apparatus itself (using compensation by force of hands, correcting spatial alignment, and/or fiddling with the knobs).

In the second approach, typically applied only briefly, the child relied upon visual feedback. The child would either start by equating number and then do repairs by means of visual feedback, or do visual feedback from the beginning.

The third approach constituted a much more adequate approach to the task. Children conceptualized the task in terms of the creation of two classes of equal weight. They drew flexibly upon a range of strategies for this purpose (including one-to-one correspondence, equate number, visual feedback, equate weight). Their pattern of application revealed two principles of strategy-selection: cognitive/motoric economy and knowledge of the limits of each strategy's valid application.

Elaboration of a new approach to the task

This analysis attempted to capture how this first approach, based on equal number with repairs restricted to spurious manipulations of the apparatus, becomes transformed into the last approach, whereby the child flexibly draws upon the range of strategies to efficiently arrive at equating of weight. In this paper, we focus on the child's knowledge of the weight variable vis-a-vis this domain and the steps by which this knowledge becomes elaborated. Again, we base our steps on distinctions found in fine-grained analysis of both macro-developmental and micro-developmental contexts of knowledge-transformation.

Extension of the variable's representation

Fine-grained analysis in both micro- and macro-developmental contexts indicates that coming to characterize the task in terms of the critical weight variable occurs in steps of surprisingly restricted extension. Whereas Siegler's (1981) model asks *do they or don't they encode weight*, I found coming to represent a variable within a domain isn't an all or nothing phenomenon.

This analysis revealed three aspects in which children need to expand their thinking in terms of weight: *a*) weights (equal or unequal) of discrete elements and *b*) the collective weight of the sets of elements in a pan and *c*) the tilt of the apparatus as an indicator of relative weight of sets of elements in each of the pans. Children frequently exhibited representation of the weight-variable in one aspect, without the others (e.g. repeatedly referring to comparative weight of the pans, while ignoring the critical distinction between weight of the discrete elements).

Extension of goal-variable coupling

Another form of extension considered by our analysis involved the scope of the context, in which the child linked the weight-variable with the goal. Again, we found steps incorporating differentiations, non-intuitive to the adult. The pathway of connecting the weight -variable to the goal came in steps corresponding to the three contexts of weight, elaborated above. For example, we observed children (throughout their protocol or at some phase of their experimental session) who represented weight in all three aspects, and yet failed to construct the relevancy of the difference for goal-attainment.

Extension of implication

The analysis also considered the pathway of implications, by which the most adequate equating-of-weights approach became elaborated. Protocols in which learning took place were the most informative.

We found that children consistently formulated the diagnostic implications before the corresponding causal implications. Although it appears obvious that the child would need to construct a new sense of what was wrong before constructing a new approach to action, the diagnostic implication frequently did not function as a constraint upon the search for an operator: the child would diagnose the problem in terms of unequal weights, and yet frequently carry out an extensive search for a repair, ignoring the weight variable.

The presence of different types of elements within the same set constituted a major obstacle for many children. Without the idea of compensation, one cannot construct a state-algorithm for placements of these sets. We observed two pathways by which children constructed implications incorporating compensation.

In one route, after repeatedly failing in the formulation of an adequate placement-algorithm, some children shifted strategically to the feedback algorithm. They attained the goal-state, by successive visual-feedback repairs. Their characterization of the goal-state, in terms of lots of little/light ones in one pan and one big/heavy one in the other, functioned as a catalyst in their construction of the compensation state-algorithm.

In the other route, children who had failed to construct the compensation implication in the context of weight-differences were able to do so in the context of size differences. (The typical rationale offered here that three of the little "make" up the big one suggest that simple composition supported the idea.) For some children, construction of the compensation idea in the size-difference context functions as a bridge for the construction of the idea in the weight-difference context. For other children, most frequently under conditions of no-feedback, the implication does not extend to the weight context.

CONCLUSIONS

This project investigated the architecture of children's developing knowledge of a physics domain. For this purpose, I took the tact taken by Inhelder's research group for the study of conceptual change: micro-analysis of learning that results from solving physics tasks. This approach enabled us to observe knowledge-fragments and pathways of connections formed, in the transformation of the preschoolers' physics knowledge-structures.

A sequence of three action-plans appeared in the micro- and macro-developmental contexts. In the first, children approached every set in the same way, placing the same number in each pan, regardless of the weight or size differences. After a pan descends, they do not adjust elements across the pans, but seek to directly fix the apparatus: by placing the apparatus into the correct alignment and gently releasing and/or hanking down on the side that rests too high. In the second, they determine initial placements through either equal number or feedback of the apparatus; they rely on feedback to determine repairs. The third action-plan integrates weight-based placement algorithms with feedback information, in a flexible system wherein the child efficiently constructs two classes of equivalent weight.

A series of micro-analyses focused on the nature and scope of the chunks and the pathway of connections manifested in the change from the initial numerically-based action-plan to the most adequate weight-based action-plan. For this purpose, we used a framework of analysis noting *a*) the scope of the domain within which the child represented a particular variable *b*) whether or not the child thought of the variable as relevant to goal-attainment; and if so, the scope of the domain in which the child assumed the mapping, and *c)* the nature and extension of diagnostic and causal implications the child had constructed.

In the spirit of diSessa's "knowledge in pieces" view of physics understanding, I found preschoolers' knowledge of mechanical equilibrium and pathway of its transformation to be comprised of remarkably small and context-sensitive fragments. Limited extension appeared in three contexts: a) representation b) conceptualization of goal relevancy and c) implication.

Children did not come to encode weight in all aspects of the domain at the same time, but first one context (e.g. relative weight of the pans) and then another (relative weight of the constitute elements). Some children, who over the course of the session came to encode weight throughout the domain, recognized the goal-relevancy of the variable in one context only. Similarly, many children did not evoke the relevant compensation structure all at once; but failed to apply it in one context (with inter-object differences in weight) and yet consistently applied it in a slightly different context (with differences in amount).

In short, the architecture of young children's physics learning appears to demand an atomistic perspective. There clearly are general structures and constructs whose states-of -evolution map onto states-of- knowledge at various levels of adequacy. However, these structures may yield a misleading account of the architecture of change, the nature of the steps by which children's knowledge becomes transformed. The young child's physics knowledge appears remarkably localized and context-sensitive. The pathway of its development consists of steps, involving a surprisingly small range of implication.

Methodologically, micro-analysis of problem-solving appears to be a fruitful means to consider the transformation of children's physics knowledge. Presenting the task, with immediate feedback and carefully selected task variations, lead to significant learning. Micro-analysis of the videotaped data allowed inference of how change entered into the child's approach and the manner by which the more adequate knowledge became elaborated.

REFERENCES

diSessa, A. (1987). *Toward an Epistemology of Physics* (Report No. 48). Berkeley, CA: Berkeley Cognitive Science Program, University of California.

Inhelder, B., & Piaget, J. (1958). *The growth of logical thinking from childhood to adolescence.* New York: Basic Books.

Inhelder, B., Ackermann-Valladao, E., Blanchet, A., Karmiloff-Smith, A., Kilcher-Hagedorn, H., Montangero, R., & Robert, M (1976). Des structures cognitives aux procédures de découverte. *Archives de Psychologie, 44*, 57-72.

Karmiloff-Smith, A., & Inhelder, B. (1974). If you want to get ahead, get a theory. *Cognition, 3*, 195-212.

Kaiser, M. K., McCloskey, M., & Profitt, D.R. (1986). Development of intuitive theories of motion: Curvilinear motion in the absence of external forces. *Developmental Psychology, 22, #* 1, 67-71.

Lawler, R. W. (1985). *Computer experience and cognitive development: A child's learning in a computer culture.* New York: Halstead Press.

Siegler, R.S. (1981). Developmental sequences within and between concepts. With commentary by Sidney Strauss and Iris Levin. With reply by the author. *Monographs of the Society for Research in Child Development, 46*(2, Serial No. 189), 75-81.

Hierarchical Problem Solving as a Means of Promoting Expertise*

Jose Mestre, Robert Dufresne, William Gerace and Pamela T. Hardiman

University of Massachusetts at Amherst

INTRODUCTION

Becoming an expert in a given domain takes substantial time and effort. This raises an interesting question: For a given individual possessing aptitude in some domain, is becoming an expert simply a function of time and effort, or can the path toward expertise be made more efficient? The research reported in this article reports on the effects of structuring the problem solving activities of novices in the domain of physics in a way consistent with the problem solving approaches used by experts. The focus of our investigation was to assess the possibility of promoting expert-like behavior among novices by constraining them to follow an expert-like approach to problem analysis.

That novices and experts store and use domain-specific knowledge in distinctly different ways is the consensus of a number of studies in such diverse fields as chess (Chase & Simon, 1973), computer programming (Ehrlich & Soloway, 1982), electrical circuits (Egan & Schwartz, 1979), and classical mechanics (Larkin, 1979). Experts tend to store information in hierarchically structured clusters related by underlying principles or concepts. When attempting to solve a problem, experts initially focus on the principles and heuristics that could be applied to solve that problem (referred to as deep structure cuing). In contrast, the knowledge base of novices is less structured and has fewer interconnections. When solving problems, novices do not focus on principles or heuristics that could be used to construct a solution strategy; rather, they focus on objects and descriptor terms in the problem (called surface features) and then look for the actual equations that could be manipulated to yield an answer (Chi, Feltovich & Glaser, 1981; Larkin, McDermott, Simon & Simon, 1980; Mestre & Gerace, 1986).

Studies in the domain of physics (Eylon & Reif, 1984; Heller & Reif, 1984) suggest that instructional approaches that impose a hierarchical, expert-like organization both on information, and on problem solving heuristics result in improved problem solving and recall performance among novices. Despite these, and other related findings current instructional practice does not emphasize hierarchical approaches to knowledge organization or to problem solving. Consequently, much of the the expert's tacit knowledge remains a secret to the novice until she or he discovers it on her own.

In the present study novices actively participated in problem solving activities which were structured to reflect our best understanding of how physics experts analyze problems. The treatment involved five one-hour sessions during which subjects solved a total of 25 classical mechanics problems using a hierarchical, computer-based, problem-analysis environment called the Hierarchical Analysis Tool. The effectiveness of this treatment

---

* Supported by National Science Foundation grant BNS-8511069. The contents herein do not necessarily reflect the position, policy or endorsement of NSF.

was compared with that of two control treatments in which novices solved the same problems using more traditional, novice-like approaches. To assess the effectiveness of the treatments at promoting shifts toward expertise, two types of tasks were administered before and after treatment: a problem categorization task (discussed in Experiment 1) and a problem solving task (discussed in Experiment 2).

DESCRIPTION OF COMPUTER-BASED ENVIRONMENTS USED IN TREATMENTS

Appreciating the findings of this study requires some understanding of the treatments that subjects received. In this section we briefly describe the architecture and functioning of the Hierarchical Analysis Tool (henceforth HAT) used in the focal treatment. We also describe another computer-based, equation-based environment used in one of the two control treatments.

The HAT is a menu-driven environment that constrains users to perform a top-down analysis of classical mechanics problems. It combines declarative and procedural information in a hierarchical framework. The environment is capable of handling the majority of problems in a typical calculus-based freshman-level classical mechanics course. The word "tool" in the name implies that the HAT can be used to facilitate the construction of a solution; the HAT does not supply answers to problems.

To analyze a problem, the user answers well-defined questions by making selections from menus that are dynamically generated by software. In the first menu, the user selects one of four general principles that could be applied to solve the problem under consideration. Subsequent menus focus on ancillary concepts and procedures, and are dependent upon the prior selections made by the user. When the analysis is complete, the HAT provides the user with a set of equations that is consistent with the menu selections made during the analysis. If the analysis is carried out appropriately, then these equations could be used to generate a solution to the problem; however, the user must still manipulate these equations to isolate the quantity asked for in the problem. In the event that the analysis is carried out incorrectly, the final equations are consistent with the user's choices, but inappropriate for solving the problem. It is important to note that the HAT neither tutors nor provides feedback to the user--it merely constrains the type and order of questions that need be considered when analyzing a problem. Figure 1 provides a sample problem and the HAT menus and selections that would appropriately analyze the problem.

A second computer-based environment, called the Equation Sorting Tool (EST), was developed for use as a control treatment. The EST was designed to be consonant with the approach taken by most novice physics students. It is a data base of 178 equations taken from a standard classical mechanics textbook. This equation data base can be sorted in three different ways: 1) by Problem Types, such as "inclined plane" and "falling bodies," 2) by Variable Names, such as "mass" and "velocity," and 3) by Physics Terms, such as "potential energy" and "momentum." By "sorted" we mean that the user can perform sequential logical "ands" to narrow down the data base to a small, manageable number of equations that might be useful for solving a problem. The EST was designed to reflect novices' problem solving tendencies in physics: they tend to cue on surface features in deciding how to attack a problem, and focus their problem solving efforts on finding the appropriate set of equations that can be manipulated to yield an answer to the problem. Further details on the design and functioning of both the HAT and the EST can be found elsewhere (Dufresne, Gerace, Hardiman & Mestre, 1987; Mestre & Gerace, 1986).

FIGURE 1. Hierarchical Analyzer Menus & Choices for Problem 1

1 Which principle applies to this part of the problem solution?

1. Newton's Second Law or Kinematics
2. Angular Momentum
3. Linear Momentum
4. Work and Energy

Please enter your selection: [4]

(B)ackup (M)ain menu (G)lossary (Q)uit (L)ist selections

2 Describe the system in terms of its mechanical energy

1. Conservative system (conservation of energy)
2. Non-Conservative system (work-energy exchange)

Please enter your selection: [1]

(B)ackup (M)ain menu (G)lossary (Q)uit (L)ist selections

3 Describe the changes in mechanical energy. Consider only the energy of one body at some initial and final state

1. Change in kinetic energy
2. Change in potential energy
3. Change in potential and kinetic energies

Please enter your selection: [3]

(B)ackup (M)ain menu (G)lossary (Q)uit (L)ist selections

4 Describe the changes in kinetic energy

1. Change in translational kinetic energy
2. Change in rotational kinetic energy
3. Change in translational and rotational kinetic energies

Please enter your selection: [1]

(B)ackup (M)ain menu (G)lossary (Q)uit (L)ist selections

5 Describe the boundary conditions

1. No initial translational kinetic energy
2. No final translational kinetic energy
3. Initial and final translational kinetic energies

Please enter your selection: [1]

(B)ackup (M)ain menu (G)lossary (Q)uit (L)ist selections

6 Describe the changes in potential energy

1. Changes in gravitational potential energy
2. Changes in spring potential energy
3. Changes in gravitational and spring potential energies

Please enter your selection: [1]

(B)ackup (M)ain menu (G)lossary (Q)uit (L)ist selections

7 Describe the boundary conditions

1. No initial gravitational potential energy
2. No final gravitational energy
3. Initial and final gravitational energy

Please enter your selection: [2]

(B)ackup (M)ain menu (G)lossary (Q)uit (L)ist selections

8 Is there another body in the system which has not been examined?

1. Yes
2. No

Please enter your selection: [2]

(B)ackup (M)ain menu (G)lossary (Q)uit (L)ist selections

9 The Energy Principle states that the work done on the system by all non-conservative forces is equal to the change in the mechanical energy of the system:

$$W_{nc} = E_f - E_i$$

According to your selections,

$W_{nc} = 0$ (Conservative system: mechanical energy conserved)

$E_f = (\frac{1}{2}Mv^2)_{1f}$

$E_i = (Mgy)_{1i}$

Please press any key to continue

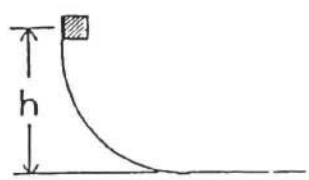
10 *** Work and Energy ***

1. Problem solved
2. Return to Main Menu to continue solution
3. Review previous solution screens

Please enter your selection:

PROBLEM 1

A SMALL BLOCK OF MASS M SLIDES ALONG A TRACK HAVING BOTH CURED AND HORIZONTAL SECTIONS AS SHOWN. IF THE PARTICLE IS RELEASED FROM REST AT HEIGHT h, WHAT IS ITS SPEED WHEN IT IS ON THE HORIZONTAL SECTION OF THE TRACK? THE TRACK IS FRICTIONLESS.



PARTICIPANTS IN EXPERIMENTS 1 AND 2

Subjects

Forty-two undergraduate students at the University of Massachusetts who had completed the first semester physics course for majors or for engineers, and received a grade of B or better, participated in this study. The subjects participated in ten hour-long experimental sessions, for which they were paid fifty dollars.

Groups

On the basis of pretest scores, the 42 subjects were divided into three treatment groups of 14 subjects each. Each of the three groups received the same 25 problems over the course of the treatment. Subjects solved five problems in each of five sessions over approximately three weeks; the treatment problems were representative of problems the subjects had

encountered in their course and covered the major topics in a beginning classical mechanics course. The "HAT-group" used the Hierarchical Analysis Tool in solving the 25 treatment problems. The "EST-group" used the Equation Sorting Tool in solving the treatment problems, while the "T-group" used a homework-style approach in solving the treatment problems and were free to refer to the textbook that they had used in their course to solve the problems.

EXPERIMENT 1: SIMILARITY JUDGMENT TASK

We designed a similarity judgment task (Hardiman, Dufresne & Mestre, 1987) in which subjects were to decide which of two comparison problems would be solved most similarly to a third model problem. Surface feature and deep structure similarity to the model problem were varied systematically, allowing us to investigate whether subjects were more likely to focus on deep structure similarity as a basis for categorization following treatment. Since the initial decision that must be made in the HAT concerns the principle to be applied, we hypothesized that the HAT-group would be more likely to focus on deep structure after treatment than either of the two control groups.

This task contained 20 items. Each task item was composed of three elementary mechanics problems, each of which was three to five lines long and contained only text (no pictures or diagrams). For each item, one of the three problems was identified as the <u>model problem</u>, while the other two were the <u>comparison problems</u>. The subjects were to indicate which of the two comparison problems they believed "would be solved most similarly" to the model problem.

A comparison problem could share different attributes with its model problem. Four types of comparison problems were designed that matched the model problem in: 1) surface features, meaning that the objects and descriptor terms that occur in both problems are similar, 2) deep structure, meaning that the physical principle that could be applied to solve both problems is the same, 3) both surface features and deep structure, or 4) neither surface features nor deep structure. These four types of comparison problems were termed S, D, SD, and N, respectively.

The comparison problems were paired such that only one of the two comparison problems matched the model problem in deep structure. This constraint led to four types of comparison problem pairs: 1) S-D, 2) S-SD, 3) N-D, and 4) N-SD. Assuming a categorization scheme based strictly on surface features, the following pattern of performance was predicted: 1) S-D: 0% deep structure choices, 2) S-SD: 50% deep structure choices (both choices are equally good in terms of matching the model problem on surface features), 3) N-D: 50% deep structure choices (either alternative is equally "bad" in terms of matching the model problem on surface features), and 4) N-SD: 100% deep structure choices (a surface feature match to the model problem will also mean a deep structure match). In contrast, assuming a deep structure categorization scheme would result in 100% deep structure choices in all four pairings.

The task was presented via computer. The subject was told to read carefully the model problem and two comparison problems, and to respond by pressing one of two keys. The items were presented in random order, with no limit imposed on time to respond. After every 5 items, the subject was given the opportunity to take a brief rest. Most subjects completed the task within 45 minutes. The same task was presented after the subjects had completed the five treatment sessions.

Results and Discussion

The performances of the 42 subjects were compared in a 3 (Treatment Groups) x 14 (Subjects/Group) x 2 (Times: pre & post) x 4 (Comparison Problem Pairings) x 5 (Model Problems) analysis of variance. The focal question was whether the HAT treatment would promote a shift toward reliance on deep structure rather than on surface features. The results indicate an affirmative response to this question. In the pre-judgment task, there were no differences between the groups, as can be seen in Table 1. However, there were significant differences among the three groups in the amount of improvement from pre- to post-treatment on the judgment task (see Table 1), $F(2,39) = 4.28$, $p=.02$.

The HAT-group was the only group to show any indications of improvement; their improvement was statistically significant, ($F(1,13)=5.20$, $p=.04$). In contrast, the mean performance of the EST-group remained the same, while the performance of the T-group declined. This result suggests that the HAT does promote a shift toward the use of deep structure, while the two control treatments do not. This shift was consistent across Comparison Problem Types, with improvements of at least 6 percentage points in each of the four comparison problem pairings. This improvement was significant for the S-D pairings (a 17% pre-to-post improvement), $t(13) = 3.12$, $p=.0324$ (adjusted for four tests), which is encouraging given that the S-D items, where surface features and deep structure are in direct competition, present the most difficulty for novices (Hardiman, et al., 1987).

Table 1: Pre- and Post-Judgment Task Percent Correct for the 3 Groups

Group	pre-treatment	post-treatment
HAT	56%	66%
EST	61%	61%
T	62%	58%
Total	60%	62%

EXPERIMENT 2: PROBLEM SOLVING TASK

To measure the effect of treatment on problem solving, two equivalent tests were constructed in the style of a traditional final exam for a freshman level classical mechanics course. Half of the subjects received one form of the test on the pre-assessment while the other half received the second form. The form not used on the pre-assessment was used for the post-assessment. The tests contained 7 problems, with 4 questions requiring the application of one physical principle for solution and the remaining 3 problems requiring the application of two principles. Subjects were given approximately one hour to solve all 7 problems.

We expected that all subjects would improve in performance from the pre- to the post-test, since all subjects would have practiced solving problems during the treatment phase. If the HAT-group was capable of adopting and applying the concept-based approach on the post-test, we might expect them to exhibit better pre-to-post improvements than the EST- and T-groups.

Results and Discussion

The tests were graded independently by two physicists. Whenever the score on an item differed between the graders, the subject's solution was

reevaluated and a score was determined by consensus. The pre- and post-test scores are shown in Table 2. All three groups increased about 10 percentage points, mainly due to improvements on the single-principle problems. Although pre-to-post improvements were statistically significant ($F(1,39)=21.25$, $p<.0001$), no one group improved significantly more than any other group. This suggests that, at least for treatments lasting a short period of time, the improvement on problem solving was primarily due to practice in problem solving in general, not to any specific treatment.

Table 2: Percent Correct (S.D.) in Pre-, and Post-Problem-Test

Group	Pre-Test	Post-Test
HAT	29.4 (20.1)	41.3 (17.5)
EST	36.4 (25.8)	44.9 (25.9)
T	31.6 (24.6)	44.4 (24.4)

Given that the improvement of the HAT-group was not significantly better than that of the two control groups, we might ask whether the HAT-group was able to use the HAT appropriately? Our data indicate the answer is no: an analysis of the HAT-group's key-stroke data indicates that subjects were able to carry out appropriate analyses using the HAT on less than half of the treatment problems. Thus, the full potential of the HAT approach for improving problem solving skills cannot be fully evaluated until we ensure that subjects adopt the approach incorporated in the HAT.

GENERAL DISCUSSION

The results of Experiment 1 indicate that the hierarchical approach to problem solving, as exemplified by the HAT, helps students to shift their decision making criteria for problem categorization from one based on surface features toward one based on deep structure. We speculate that use of the HAT promotes this shift because it highlights the importance of applying principles to solve problems by asking subjects to select the applicable principle in the first menu they encounter. Even if the user is not able to answer all of the subsequent questions in the analysis correctly, the principle to be applied to obtain a solution may still be recognized as primary.

However, the current implementation of the hierarchical approach did not lead to significantly more improvement in problem solving than a "homework style" approach. Assuming the hierarchical approach is a potentially powerful tool for improving problem solving, there are two main reasons why the HAT treatment failed to yield more significant improvements in problem solving. First, subjects were not able to internalize the approach implicit in the HAT, since they had no way to gauge whether or not they were using the HAT appropriately. It appears that feedback and coaching are necessary ingredients to help novices assimilate the HAT's expert-like approach (Collins, Seely Brown & Newman, in press). Second, the treatment was relatively short, and therefore it is unrealistic to expect dramatic reorganization of declarative and procedural knowledge after using the HAT for only 5 hours.

Two implications appear to stand out. First, hierarchically structuring the problem analysis activities of novices holds promise for promoting expert-

like behavior among novices. However, simply partaking in expert-like problem solving activities is not sufficient to promote dramatic improvements in a complex task such as problem solving--we need to guarantee that novices actually adopt the hierarchical approach before we can evaluate its full potential. Second, problem solving assessments may not be the most sensitive for measuring modest shifts toward expertise. Other measures, such as problem categorization, appear to be more sensitive assessments of subtle shifts toward expertise.

REFERENCES

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.

Collins, A., Brown, J.S. & Newman, S.E. (in press). Cognitive apprenticeship: Teaching the craft of reading, writing and mathematics. In L. Resnick (Ed.), Cognition and Instruction: Issues and Agendas. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Dufresne, R., Gerace, W., Hardiman, P. & Mestre, J. (1987). Hierarchically structured problem solving in elementary mechanics: Guiding novices' problem analysis. Proceedings of the Second International Seminar on Misconceptions and Educational Strategies in Science and Mathematics (Vol. III, pp. 116-130). Cornell University, Department of Education: Ithaca, NY.

Egan, D.E. & Schwartz, B.J. (1979). Chunking in recall of symbolic drawings. Memory & Cognition, 7, 149-158.

Ehrlich, K. & Soloway, E. (1982). An empirical investigation of the tacit plan knowledge in programming. Research Report #236, Department of Computer Science, Yale University.

Eylon, B.S. & Reif, F. (1984). Effect of knowledge organization on task performance. Cognition & Instruction, 1, 5-44.

Hardiman, P.T., Dufresne, R. & Mestre, J. (1987). Physics novices' judgments of solution similarity: When are they based on principles? Proceedings of the Second International Seminar on Misconceptions and Educational Strategies in Science and Mathematics (Vol III, pp. 194-202). Cornell University, Department of Education: Ithaca, NY.

Heller, J.I. & Reif, F. (1984). Prescribing effective human problem solving processes: Problem description is physics. Cognition & Instruction, 1, 177-216.

Larkin, J. H. (1979). Information processing models in science instruction. In J. Lochhead & J. Clement (Eds.), Cognitive Process Instruction. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Larkin, J.H., McDermott, J., Simon, D. & Simon, H. (1980). Models of competence in solving physics problems. Cognitive Science, 4, 317-345.

Mestre, J.P. & Gerace, W.J. (1986). Studying the problem solving behavior of experts and novices in physics via computer-based problem-analysis environments. Program of the Eighth Annual Conference of the Cognitive Science Society (pp. 741-746). Hillsdale, NJ: Lawrence Erlbaum Assoc.

COLLABORATIVE COGNITION

Barbara A. Fox and Lorraine Karen
Linguistics Department
University of Colorado, Boulder

INTRODUCTION

This paper attempts to show that a central function of human tutorial dialogue is what we will call public, or collaborative, cognition, whereby the tutor, with help from the student, makes overt the usually private cognitive processes which lie behind problem solving. In this way the tutor and student together produce the processes which the student will eventually be called on to do by him or herself (for example, on an exam). This phenomenon has not, to the best of our knowledge, been documented before for any adult interaction and has not been addressed in the tutoring literature (something like it has been discussed in the language development literature; see Ochs & Schieffelin, 1983; Vygotsky, 1978).

We provide support for this hypothesis about tutoring by analyzing two important phenomena in our tutoring data: (1) use of analogies, and (2) reading word problems. We have found it especially useful to concentrate on these issues since they have both been the source of some controversy in cognitive science.

We also briefly discuss the mechanisms which we believe produce the externalization of internal processes.

BACKGROUND

The data on which this study is based were collected as part of a larger project on human tutorial dialogue. For this larger study, we brought together experienced tutors (graduate students from Physics, Chemistry, Computer Science and Math, at the University of Colorado) and students actively seeking tutoring (most were undergraduates). We then video-taped each tutor-student pair in three contexts: face-to-face, terminal-to-terminal, and student-to-computer (where the student thought s/he was using a computer tutor but in fact was interacting with his/her human tutor). Only the first context is relevant to the present report.

In this passage, when the analogy is first produced, the student appears not to understand (she doesn't respond), but after the tutor suggests one of the pieces internal to the base ("it's got mgh"), the student is able to formulate another internal piece ("it's got potential energy up here"). Now the relevant structure of the base has been established. The tutor then goes on to provide two relations which map the base to the target ("this guy describes what kind of gravitational field there is" and "this guy happens to be one meter from"), and the student is able to produce the final mapping ("he's like the stone"); the analogy now appears to be completely constructed. In this example, then, the tutor and student publically go through the cognitive process of first producing the relevant internal structure of the base, then mapping this structure to the structure of the target: this passage is thus a prototypical example of what we have called collaborative cognition. The two participants work together (obviously the tutor leads the work) to make explicit what might otherwise go on "in the head" of just one participant.

WORD PROBLEMS

In recent work, Dellarosa (1986), and Dellarosa, Kintsch, Reusser & Weimer (1987) have shown that young children's difficulty with arithmetic word problems lies in their lack of linguistic sophistication, their inability to go from the linguistic description (the word problem itself) to the desired conceptual representation. Our research both challenges supports this finding and challenges it. It supports the finding in that the students in our study clearly had difficulties with what one might loosely describe as "understanding what the question was really asking for"; that is, they could not easily go from reading the problem to setting up the answer, even if they in fact knew how to solve such a problem. This indicates that Dellarosa and Kintsch are exactly right; what causes difficulty is the mapping, if you will, between the text and the conceptualization.

However, given that our students are college-level, taking very difficult courses, we cannot assume that the trouble arises from a lack of linguistic sophistication; by all measures, these students are fully developed linguistically. Why, then, do college students find word problems problematic? Our

answer is that the correct interpretation of a word problem (that is, discovering the correct unknown and knowns) is a matter of convention (schema-driven, in a sense, with critical key words and phrases), and each student must learn these conventions to solve the problems at hand. Moreover, not only are they a matter of convention, but they are a field-specific matter of convention, such that one must learn the conventions for each discipline (perhaps even for each new course in a university curriculum). In order to know what the problem is "really" asking for, one must be exposed to the conventions of problem interpretation.

The tutors that we have studied take interpretation as one of their central tasks; that is, one of their major goals appears to be to teach their students how to read the problems. This is not to say that there is no conceptual level instruction going on; obviously there is. But both participants appear to recognize that this conceptual knowledge will not get a student through an exam if s/he cannot go from the problem to the conceptual level.

The fact that tutors engage in this behavior is supported by examples like the following, wherein the tutor and student explicitly address what the problem is asking for.

T: [reading problem] How close must two electrons be if the electric force between them is equal to the weight, of either at the earth's surface.

.

.

S: So this is the formula I picked out. What they want is the force.Right?
T: Right.
S: No they want how close they should be.
T: Right, but that's what that is, right?
S: Yeah.

The following tutor comments further demonstrate the general concern:

T: [reading problem] Given three point charges, fixed at three corners of a square. Find the electric field and tendency vector ee, at the corner pee, with no charge, this is exactly like the problem we just did.
S: Right.
T: And show the direction with a carefully drawn arrow.

So, you'd have to find the angles of it.
S: Aha.

It should be clear from these examples that tutors are concerned with the issue of interpreting the problem. Now we need to ask how the tutors go about this task. Our claim is that they go about this task by making explicit, making public, some version of their own thoughts when they solve problems, e.g. what sorts of questions they ask themselves at a given point, what certain kinds of key phrases they look for, and how to judge if a problem is "the same" as some other problem. In other words, the tutors make public the schemas they believe are necessary for understanding certain kinds of problems.

A clear example of this phenomenon follows.

T: [reading problem] What is the speed of a three hundred and fifty EV electron. Okay.
(1.2)
T: So the main thing here, I mean, when you look at that, what is electron volts, what kind of a what are we talking about.
.
.
T: So they want you to relate speed to energy.
S: Oh, okay.
.
.
T: When you look at this, you know you got this 350 electron volts, and you go- and you always go, on my God what what is an electron volt, and then if you can any way fool around with the units, to figure out well what is it I'm talking about, you know, and you go well it's q times v and you look over your equation and you go okay well that's, work and energy are the same thing.

In this passage the tutor models for the student how to go from what is explicitly given in the words of the problem to other formulations of the same facts, to arrive at the "real" question the problem is posing. In her final comments she even gives a *general* strategy for how to get from the superficial

structure of the problem to what "I'm talking about." She models this process by making overt some version of her own internal cognitive steps (what this "version" is will be clarified below.)

CONCLUSIONS

We have demonstrated in this study that one of the main tasks of a tutor is to make public processes which might otherwise take place privately, within the "head" of one person. This modeling of normally private problem-solving behavior is invaluable for students because it allows them to participate in, to whatever extent they are able, a process which is otherwise inaccessible to them and which they will soon have to be able to do alone.

It is important to note at this point that in making their internal processes explicit the tutors are not merely going through the steps they would go through if they were working a problem alone (although this does happen on occasion); clearly those procedures are too compiled to be of much use to a novice. Nor do they (usually) make explicit the internal processes of the current student; clearly that would be of no learning value to the student. So what the tutors do is engage in making explicit the ideal cognitive processes of a good novice, not of an expert and not of the exact student sitting before them (Anderson et al, 1986). However, the tutors do this not by imagining an abstract good novice, but by projecting the particular student involved into the role. That is, they display how **this particular student** would ideally behave faced with this problem. This tactic allows the student to see some extended version of his/her own cognitive processes successfully solving problems. In this way the student is maximally encouraged, inasmuch as s/he sees him/herself at a level beyond the current level.

REFERENCES

Anderson, J., Boyle, C., Corbett, A., & Lewis, M. (1986). Cognitive modelling and intelligent tutoring. Carnegie Mellon Tech Report.

Clement, J. (1983). Observed methods for generating analogies in scientific problem solving. Presented at the annual meeting of the American Educational Research Association, Montreal, Canada.

Dellarosa, D. (1986). A computer simulation of children's arithmetic word problem solving. *Behavior Research Methods, Instruments, and Computers, 18*, 147-154.

Dellarosa, D., Kintsch, W., Reusser, K., & Weimer, R. (1987). The role of understanding in solving word problems. University of Colorado, Boulder, Mss.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science, 7* (2), 155-170.

Gentner, D. (1987). Analogical inference and analogical access. University of Illinois Tech Report UIUCDCS-R-87-1365.

Holyoak, K. (1985). The pragmatics of analogical transfer. In G. Bower (Ed.), *The psychology of learning and motivation* (Vol. 1). New York: Academic Press.

Ochs, E. & Schieffelin, B. (1983). *Acquiring conversational competence.* Boston: Routledge and Kegan.

Vygotsky, V. (1978). *Mind in society.* Cambridge: Harvard University Press.

Explorations in Understanding How Physical Systems Work

Barbara Y. White & John R. Frederiksen
BBN Laboratories

ABSTRACT

This paper presents a theory of how to enable people to understand how physical systems work[1]. Two key hypotheses have emerged from our research. The first is that in order to understand a physical system, students need to acquire causal mental models for how the system works. Further, it is not enough to have just a single mental model. Students need alternative mental models that represent the systems behavior from different, but coordinated, perspectives, such as at the macroscopic and microscopic levels. The second hypothesis is that in order to make causal understanding feasible in the initial stages of learning, students have to be introduced to simplified models. These models then get gradually refined into more sophisticated mental models. We will present a theory outlining (1) the properties of an easily learnable, coherent set of initial models, and (2) the types of evolutions needed for students to acquire a more powerful set of models with broad utility.

INTRODUCTION

We have been creating an intelligent, computer-based, learning environment that helps students to learn about the behavior of electrical circuits (Frederiksen & White, 1987; White & Frederiksen, 1986, 1987). The issues that have confronted us are relevant to the understanding of any physical system, including both designed and naturally occurring systems. By presenting the evolution in our attempts to foster understanding, we can raise some important issues concerning what it means to understand, and how to go about facilitating understanding.

"Causality", "mechanism", and "purpose" are perspectives that people often spontaneously impose on physical systems in order to understand how they work. These perspectives contrast with the type of constraint-based, quantitative reasoning that physicists often use in modelling physical systems, and that is presented to students in physics class and textbooks. This lack of connection between students' intuitive notions of causal mechanisms and quantitative circuit theory is, we believe, one reason why students typically have so much difficulty understanding the behavior of electrical circuits.

Our initial hypothesis was, therefore, that students would learn circuit theory more readily if it were introduced in the form of qualitative, causal models of circuit behavior. Causal models can simulate domain phenomena, and can provide a means of linking mathematical formalisms to these phenomena. In addition, they can introduce students to the basic concepts, laws, and causality of the domain. When internalized in the form of mental models, they can enable students to mentally simulate and to explain domain behavior. Such mental models can be used in solving a wide range of interesting problems, such as predicting circuit behavior, and designing and troubleshooting circuits. The computer system we built therefore incorporated causal models that could visually simulate and verbally explain the causality of circuit behavior. The system was designed to provide an external representation of the mental models that we wanted students to acquire.

[1]This research was funded by the Army Research Institute, under contract MDA-903-87-C-0545.

Causal models can embody different perspectives on system operation. For instance, such models can reason about circuit behavior at what might be called the macroscopic, or phenomenological level. That is, they can allow reasoning about circuit behavior through the application of a set of laws that govern the distribution of voltages and currents within the circuit. One can imagine, for example, closing a switch and hearing an explanation of how that causes voltages to be applied to devices within the circuit, and how that in turn causes a current through a device such as a light bulb, which causes it to be on. Alternatively, causal models can represent the behavior of circuits at the more microscopic level. For instance, one can imagine seeing how electrical forces within a circuit can cause mobile, charged particles to be redistributed when, for example, a switch is closed. Finally, causal models can take the form of functional models, which reason about the purpose of the circuit, and how subsystems within the circuit interact to achieve that purpose. Here, instead of reasoning about devices as potential conductors or sources of voltage, one reasons about devices as receivers, transformers and transmitters of information, and the causal propagation is in terms of information flow, instead of in terms of electrical charge (White & Frederiksen, 1987b). The question for instruction then is: which of these types of causal models should one start with and why? Further, why having picked one of them, are there any compelling reasons for introducing the others?

PHENOMENOLOGICAL MODELS

We began by conducting studies of experts' reasoning about circuit behavior in the course of solving problems such as troubleshooting. We found that experts primarily reason about device states, and how a change in the state of one device can cause changes in the states of other devices. When the circuits dealt with were simple electrical circuits such as those found in an automotive electrical system, experts employed electrical concepts in predicting how a change in one device's state would propagate its effects to other devices within the circuit. This electrical reasoning incorporated the idea of causal interactions among devices, which are due to the effects of changes in conductivity of one device on voltages applied to other devices within a circuit. The experts reasoned solely about whether voltages would be present or absent across devices, not about incremental changes in those voltages or about their quantitative values.

To model this type of reasoning, the model we created reasons about gross aspects of circuit behavior, such as whether or not there is a voltage drop across a device. For example, in a circuit containing a battery, a switch, and a lightbulb, the expert reasons that (1) when you close the switch, that changes the conductivity of the switch causing a redistribution in the voltages within the circuit, (2) so that now there is a voltage drop across the lightbulb, whereas before there was not, (3) so the lightbulb goes from being off to being on. In creating this model, we initially adopted what might be called a "device centered" view of casuality. That is, reasoning about the effects of a change in circuit conductivity was done from the perspective of the individual devices in the circuit. For example, each device within a circuit would ask "What effect would closing that switch have on my behavior?". So, for instance, the lightbulb in this case would say "Before the switch was closed, I didn't have a good feed path to the battery, and now when the switch is closed, I do have a good feed path, and I also have a good return path, so I have a voltage drop across me, so I will be on". We found that such a model is easy to learn and is useful for introducing basic ideas, like the concept of a circuit.

One of the drawbacks of the device-centered model, however, is that it doesn't map readily to basic circuit laws (such as Kirchhoff's Laws) which are important in quantitative circuit analysis. These laws may be thought of as processes which govern the distribution of voltages and currents within an electrical

circuit. We therefore went on to consider an alternative perspective (Forbus, 1985) in which the concept of "process" is regarded as central to understanding how systems work. For instance, you could reason that when you close a switch, there is a change in the switch's conductivity, which initiates a "voltage redistribution process". Instead of looking at what is going on from the perspective of each device, this redistribution process is run whenever a device changes state. Then, all devices within the effected part of the circuit simply ask whether the voltage drop across them has changed, and if it has, they alter their states appropriately. The voltage redistribution process takes the form of a set of qualitative rules such as, "If you have a circuit with an open in it, the only voltage drop in that circuit will be across the open. Whereas, if you have a circuit that is a complete conductive loop, there will be voltage drops across resistive devices in that circuit." These rules are, in effect, qualitative expressions of the circuit laws of quantitative circuit theory. Students can be easily motivated to develop this new, "process" perspective by presenting them with circuits (e.g., long series circuits) for which the device centered approach is extremely tedious and inefficient. For these circuits, the process-centered model represents a much-needed simplification.

Each of these alternative phenomenological models represents a different perspectives on circuit causality. Thus, in addition to being introduced to some fundamental concepts about circuits, such as voltage, conductivity, and the basic idea of circuit, students learn that there is more than one way to approach modelling circuit behavior that is consistent with the fundamental concepts. In each approach, we imposed a causality onto the behavior of circuits, such as a change in conductivity can cause a change in voltage, and a change in voltage can cause a change in device state. This causality maps nicely to ideas about electrical force and the behavior of charge carriers within a circuit, as well as to quantitative circuit analysis. However, there are serious limitations to only presenting student with such phenomenological circuit models. In particular, when exposed to these kinds of causal models, students would only have shallow answers to important questions such as, "What is a voltage drop, and how do voltages get redistributed within a circuit?" Beyond the set of rules for determining the distribution of voltages, students have no deeper sense of what is going on in a circuit. One of the reasons for this shallowness (which is also a characteristic of quantitative circuit models) is that in creating the voltage redistribution process, we had used qualitative versions of the steady state circuit laws, i.e., Ohm's Law and Kirchhoff's Voltage Law. Unfortunately, if you use steady state laws to determine what the next steady state will be, you have not reduced the process to fundamental principles like F=ma (predicting the effects of electrical forces on the movement of charged particles).

REDUCTIONISTIC, PHYSICAL MODELS

We decided that we needed to unpack the voltage redistribution process via a dynamic physical model. This model would introduce a simple causal mechanism that would drive the process. The causal mechanism would be based on what diSessa (1983) refers to as phenomenological primitives, i.e., things that you can experience with your own body, such as pushes and pulls and a sense of equilibrium. Furthermore, it would be based on more fundamental physical laws such as F=ma and Coulomb's Law, rather than on the steady state circuit laws. Another important property of this model would be that by watching this physical mechanism run, the steady state circuit laws would emerge. In other words, it would be a model based on Coulomb's Law, and Ohm's and Kirchhoff's Laws would be emergent properties of the physical process embedded in the model.

It is interesting to note that when we surveyed physics text books, electrical engineering text books, and technical text books that are used in training electrical technicians, we were unable to find such a

dynamic physical model. In two advanced physics text books, we did find a mathematical derivation of how you can start with F=ma and get Ohm's Law, but these derivations still focused on reasoning directly about the properties of circuits in a steady state, not about transient phenomena. The fact that we didn't find such a mechanism presented in any of the text books suggests that, from the perspective of physicists, either it is unnecessary to develop such a model of transient phenomena within circuits, or such dynamic models are so complex as to confuse rather than enlighten. However, while such transient models are not available for electricity, they are fairly common in the study of gas diffusion and heat flow. Moreover, while physicists may accept abstract, algebraic derivations of circuit laws without any necessary link to a transient, physical model, in the initial stages of learning most students find that mathematical abstractions and algebraic derivations make sense only after the domain is understood in causal terms. If a dynamic model is not supplied, they will attempt to build their own, or to interpret the circuit laws in terms of their prior conceptions of electricity.

We therefore focused on creating a model that would embody a simple mechanism which would help people to understand the origins of circuit behavior such the voltage redistribution process. The model is based on a simplification of Coulomb's law in which we quantize distance by dividing resistors up into little slices. We also ignore surface charges and regard the resistor as essentially one dimensional. Within each slice of the resistor, we represent the charge of that slice. For example, the middle slice (with the holes filled with minuses) shown in Figure 1 is neutral, the one on the right has a negative charge, and the one on the left has a positive charge. These circles and minuses are not meant to represent individual atoms and electrons. They are just a representation of the charge of that slice. If you prefer, you can think of them as being replaced with a number, so the middle slice has a charge of 0, the right is -5, and the left slice is +5. Now if you put two of these slices next to one another, and if there is a difference in their charge, then there will be an electrical force exerted on charged particles within the two adjacent slices. This can be thought of as due to the negative charges repelling one another and the positive charges attracting the negative charges. These forces will accelerate the mobile charges (the minuses), causing them to migrate (i.e., be redistributed) from the relatively more negatively charged slice to the more positively charged slice until both slices have the same charge.

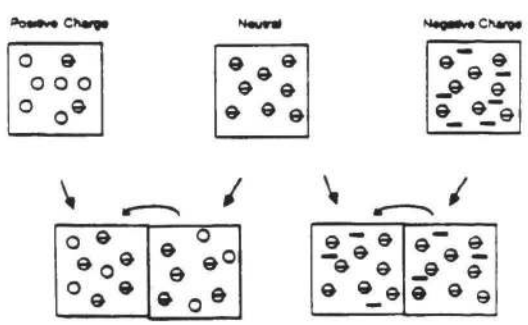


Figure 1. Slices of a resistor.

We can increase the resistance of a resistor by putting more and more of these slices next to one another. We model a battery by simply saying that it is a device which maintains a constant positive charge on one of its terminals, and a constant negative charge on the other of its terminals. Then, for example, if you attach the negative terminal of a battery to one end of a resistor a resistor as in Figure 2 (so you don't have a complete circuit), you can watch what happens over time. Within the model, time is also quantized. Within each time interval, a certain proportion of mobile charges will be redistributed, the

actual number depending on the difference in charge between the two adjacent slices. As this model runs, one can see that on each time increment, adjacent slices go part way towards reaching equilibrium. In this way, one can watch the system settle down into a final, steady state. In this case, there will be no voltage drop across the resistor and the source voltage will be developed across the open segment of the circuit. Furthermore, if you put together a complete circuit, like the one shown in Figure 3, and let the process run until it reaches a steady state, you see how you get Ohm's Law and Kirchhoff's Laws from a system that behaves in accordance with Coulomb's Law. This is a nice example of how you can start with some very simple ideas about attraction, repulsion, and equilibrium, and show how a system that operates according to a mechanism based on these ideas will settle down into a steady state that obeys Ohm's Law and Kirchhoff's Laws.

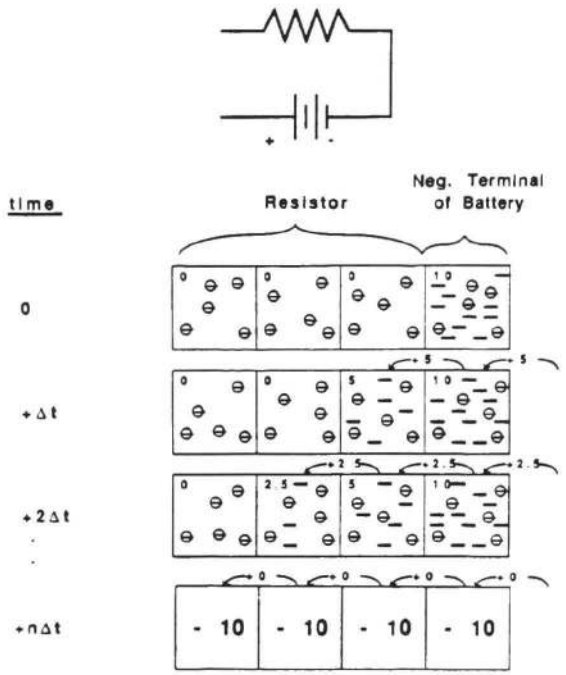


Figure 2. An open circuit.

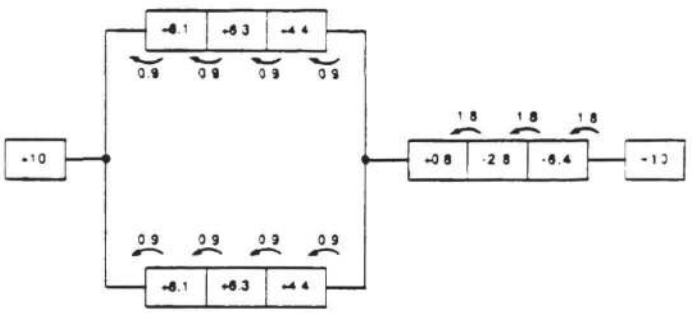


Figure 3. A series-parallel circuit.

This model makes the qualitative models described earlier "make sense". It provides a sense of

mechanism, so that the macroscopic models, combined with this unpacking of the voltage redistribution process, provide a coherent, although simplified, view of how circuits work. This model is, however, far from "complete". Firstly, there is no model provided to explain how electrical force gets produced by a battery. Secondly, the holes and minuses are not meant to represent individual atoms and electrons. Instead they provide a representation of the charge of each slice of a resistor. And, unless you actually represent the individual electrons and animate them by showing them moving and bumping into atoms, you do not get a good sense of what resistance is. So, one could hypothesize that we need to unpack one more level, and show the behavior of atoms and electrons. However, we argue that this is unnecessary. People already have good intuitions about resistance. The hard and key ideas are voltage drop and the process by which voltages get redistributed. This model represents that concept and process succinctly.

By interacting with such models, students can also come to appreciate some important distinctions among physical theories: The laws that are used to describe the steady state are usually redundant. They are, if you will, multiple constraints on the configuration of charges in the steady state. Given their redundancy, if several statements are given, one can derive the remaining ones (this is an example of constraint-based reasoning). In this way, students can be introduced to constraint-based reasoning in a qualitative form as a precursor to its algebraic forms. In general, students can come to appreciate epistemological distinctions among forms of physical theories and ways of reasoning, and learn how they can exist side by side as alternative ways of viewing the physical world.

Another interesting property of this process model emerged when describing it to physicists. We would say, "This is a diffusion model which represents a first approximation to what happens in electrical circuits. It is the kind of model that you might use, for example, to explain heat flow to someone." The physicist would say, " Oh yes, you could use it to explain what happens in gases as well". These discussions implied that there are a small number of key processes, like this diffusion process. These key processes are understandable because they can be grounded in phenomenological primitives (i.e., can introduce a simple mechanism that relates to things you can do with your own body). They also have the property that they can be used to help students understand a wide range of physical domains. Thus they can play a key role in enabling students to learn.

CONCLUSIONS

To summarize the viewpoint that has evolved in this research: We hypothesize that students need to start with a dynamic, physical model that provides a sense of mechanism and introduces domain causality. Such models are, however, not particularly useful for problem solving. Thus, in addition, one needs to acquire more abstract, behavioral models that are reasoning at a more macroscopic level. At the level, for example, of switches closing causing lights to go on and off. These kinds of models are useful not only for understanding, but also for problem solving, and need to be introduced in both qualitative and quantitative forms. However, causality and mechanism are not the only perspectives that people employ to understand how systems work. There is another perspective that people often impose on physical systems, and that is the perspective of "purpose": What is the system designed to do, and how do the subsystems within it work together to achieve that purpose? This represents an important way of coming to understand systems, particularly in the case of designed artifacts, and we have been doing further research concerning how to introduce functional models into our intelligent learning environment.

The intelligent learning environment incorporating a progression of qualitative models has been tried out with seven subjects. It was successful in all seven cases in enabling novices to mentally simulate and to troubleshoot circuits. When subjects were not provided with a version of the reductionistic, physical model, they spontaneously invented their own reductionistic explanations which were incorrect and usually inconsistent with the causality embedded in the macroscopic models (White & Frederiksen, 1987b).

In helping students to acquire this set of mental models, we have the same objectives that physicists have in evolving such models: We want students to possess a set of coherent models that will enable them to predict and understand what will happen across a wide range of domain phenomena. In the case of electrical circuits, the models we have described allow students to mentally simulate circuit behavior from different, but coordinated, perspectives. For instance, the causality of our microscopic models is consistent with that of our macroscopic models of circuit behavior. Students can employ this set of models to make predictions about circuit behavior, as well as to design and troubleshoot circuits. The models thus provide a coherent, linked, knowledge structure that can generalize across a range of situations to solve problems and to generate explanations.

REFERENCES

Brown, J.S. & deKleer, J. (1985). A qualitative physics based upon confluences. In Bobrow, D.G. (Ed.). Qualitative Reasoning about Physical Systems. Cambridge, MA: MIT Press.

diSessa, A. (1983). Phenomenology and the evolution of intuition. In D. Gentner & A. Stevens (Editors), Mental Models. Hillsdale, NJ: Lawrence Erlbaum Associates.

Forbus, K.D. (1985). Qualitative process theory. In Bobrow, D.G. (Ed.). Qualitative Reasoning about Physical Systems. Cambridge, MA: MIT Press.

Frederiksen, J., & White, B. (in press). "Intelligent tutors as intelligent testers." To appear in Diagnostic Monitoring of Skill and Knowledge Acquisition. N. Frederiksen, R. Glaser, A. Lesgold, & M. Shafto (Editors), Lawrence Erlbaum Associates, Hillsdale, New Jersey.

White, B., & Frederiksen, J. (1986). "Intelligent tutoring systems based upon qualitative model evolutions." In the Proceedings of the Fifth National Conference on Artificial Intelligence, Philadelphia, Pennsylvania.

White, B., & Frederiksen, J. (1987a). "Qualitative models and intelligent learning environments." In AI and Education. R. Lawler & M. Yazdani (Editors), Ablex Publishing Corporation.

White, B., & Frederiksen, J. (1987b). "Causal Model Progressions as a Foundation for Intelligent Learning Environments." Report No. 6686, BBN Labs, Cambridge, MA. To appear in Artificial Intelligence.

Instructional Strategies for a Coached Practice Environment

Susanne P. Lajoie, Gary M. Eggan, and Alan Lesgold[1]

Learning Research and Development Center
University of Pittsburgh
Pittsburgh, PA 15217

1. Our coached practice environment, Sherlock, was a massive undertaking and we would like to acknowledge: Jaya Bajpayee, Marilyn Bunzo, Dennis Collins, James Collins. Richard Eastman, Drew Gitomer, Bob Glaser, Bruce Glymour, Sherrie Gott, Linda Greenberg, Debra Logan, Maria Magone, Thomas McGinnis, Sandy Stanley, Arlene Weiner, Richard Wolf, Laurie Yengo, and the officers and personnel of the 1st and 33rd Tactical Fighter Wings at Langley and Eglin Air Force Bases.

The work was done as part of subcontracts with HumRRO and the Universal Energy Systems, who were prime contractors with the Air Force Human Resources Laboratory. Many of the tools that permitted us to efficiently conduct this project were put in place by earlier efforts and foresight of the Office of Naval Research.

Instructional strategies for a coached practice environment

Susanne P. Lajoie, Gary M. Eggan, and Alan Lesgold
Learning Research and Development Center
University of Pittsburgh

Better understanding of proficency in complex real-world tasks is needed, both for assessment of skill and for training. As part of a program of research on assessing and training complex problem solving, we have developed a computer program that provides coached practice for an electronics troubleshooting job. Sherlock was developed for first-level airmen who diagnose and repair navigational equipment for the F-15 aircraft. It provides trainees with realistic fault isolation problems in a simulation of the United States Air Force's F-15 manual avionics [2] test station. Avionics units are attached to a test station via a test package that adapts the unit to the test station. The test station simulates the aircraft and how the unit would work on the aircraft and measures unit performance. However, the test station is complex and is prone to failures itself. Thus, often the trainee must troubleshoot faults in the test station before being able to diagnose and repair avionics components. Troubleshooting the test station is the most complicated aspect of the job since there is little training in this area prior to on-the-job experience, and job supports are incomplete. Sherlock was developed for first level airmen, as a supported practice environment for learning troubleshooting techniques that could be applied in the real world. There are several instructional strategies that guided Sherlock's development, each of which were driven by our empirical cognitive task analysis research in this field (Lesgold et al., 1986).

Background. We assume that our trainees are active learners. They are coached as needed while solving problems, thereby presenting opportunities to construct new knowledge that is linked to the mental state at which it will be needed. Sherlock has a **holistic procedural training** emphasis, which means that supported practice of the overall troubleshooting process is provided rather than practice of discrete knowledge skills. People learn to do complex cognitive procedures by doing them. When a trainee[3] cannot handle the whole performance, he is supported through the parts he cannot handle, via reminders and hints, but at the same time he is still enacting the whole target cognitive performance. Core mental models are not explicitly revealed but are discussed and elaborated when an individual's performance indicates that explicit help is needed in a particular part of the overall target performance. Much of our tutoring simply stimulates a technican's interrogation of the problem space rather than provide explicit guidelines on what to do next. An example of how interrogation is fostered would be the following hint delivered by our tutor to a trainee working on a problem: *"What relays in the A1A1A13 are being set by this code? How can you test this card to determine if the correct relays are set?"* If such a hint were insufficient a more explicit hint would be provided, such as, *"You should test pins 41, 42, and 46 with respect to ground (pin 51) and see if the data signals are good. "*

Sherlock presents 34 problems based on our concept of "effective problem spaces" (see Lesgold, Lajoie, Logan and Eggan, in prep.). These are subgoal lattices of plans and actions that an individual might make while troubleshooting. Both expert and novice moves are included in the representation, thereby predicting ahead of time most of the steps that a technican might make on a particular problem. At each decision point or node in the effective problem space, available hints

2. Avionics refer to electronics equipment for aircraft navigation. The F-15 is a tactical fighter plane.

3. Most, but not all, of our subjects were male. We use the masculine form only to simplify sentence structure.

focused on relevant curriculum issues as well as what the appropriate actions, test outcomes, conclusions, and next best moves would be at that point. While there was no unique best path or sequence for solving a problem, there were definite expertlike ways of troubleshooting. Proficient troubleshooters collect appropriate information to confirm or disconfirm a specific hypothesis regarding fault location. The effective problem space accommodates different levels of expertise in planning. Each subgoal node has one or more expert ways to confirm or disconfirm a particular hypothesis. Less skilled individuals are tutored in these expert moves. Usually this involves hints provided on demand, but Sherlock also may intervene to limit time wasted repeating moves or following hopeless dead ends.

Every plan or action in the problem space has as many as four types of hints, for the choices of which action to take, what outcome to expect and how to interpret it, what conclusions to draw, and what to do next. For each hint type, there are five levels of explicitness. Trainees receive these hints when they ask for help or show that they are having difficulty with particular curriculum issues. How explicit the hint is depends on how well the tutor expects the trainee to perform at that point. The performance expectations are based on a student competence model that is updated after each problem. Only a small portion of the tutor design is presented here since it has been reported elsewhere (see Lesgold, Lajoie, Bunzo & Eggan, 1988).

An Integration of Theories

Glaser (1985) and Resnick (1985) have called for an integration of several kinds of theories to guide instruction: (a) a theory of competence that describes the skilled performance that we hope to evoke in the learner; (b) a theory of acquisition that describes how people construct knowledge and gain skill; and (c) a theory of intervention that specifies how to activate the learner's acquisition processes. These three theories were considered in our tutor's design.

A Theory of Competence. The development of Sherlock was driven by a cognitive task analysis of an avionics electronics troubleshooting specialty (see Lesgold et al, 1986). We derived curriculum issues for the tutor from the cognitive competencies found to characterize troubleshooting proficency. Specifically, we had looked for differences between trainees who were acquiring their skills quickly and those who were not (based both on supervisor ratings and performance of subjects on difficult diagnosis tasks). Proficient troubleshooters were skilled at generating and testing their fault isolation plans, as well as using the appropriate methods to resolve their hypotheses. These more-skilled performers were also better at the overall troubleshooting process and at using their strategic knowledge and operational methods for solving problems. The curriculum we derived from this analysis reflects 82 aspects or components of skilled performance in three major categories: (a) procedural knowledge of basic troubleshooting procedures, such as setting up the test station correctly; (b) declarative or operational knowledge of measurement-making, such as knowing which test equipment to use and when to use it; and (c) strategic knowledge, such as serial tracing through circuit paths, space splitting, and device and system understanding.

We concluded that it is better to train specific situated competence holistically, permitting transfer to come from concepts and procedures that are common or that can be generalized from experience in multiple situations, rather than to teach abstractions that a trainee might not be able to map to various situations. Rather than teach any one curriculum issue in isolation, there were often several issues that were relevant to plans or actions in a troubleshooting problem.

Hoping to make Sherlock an expert teacher, we relied on an interesting body of research into pedagogical competence. Leinhardt (1986) examined how math teachers organize their knowledge and how they transmit this organizational structure to their students. Expert math teachers provide

not only clear presentations of a new mathematical procedure but also several contexts in which it is applied. Rehearsal is provided for both the skill and understanding in many contexts so that students can see the conditions under which this new procedure applies.

In developing Sherlock we hoped to replicate this teaching expertise. **Specific domain expertise** is necessary in a tutor. For a system to be able to shape performance toward expertise, it must reflect, in its coaching, the expertise of real workers who have learned to do the job, not just the expert designer's model of the work environment or the job as represented by doctrine. Our expert, Gary Eggan, was a "worker" as well as a "teacher" who had worked in the job as an enlisted airman and later become a technical advisor to airmen doing the job. He provided us with a rationale for problem selection and with clear representations of the problem space for each problem, including likely expert and novice paths to solution. Thus several solution paths were mapped out along with coached practice for each path. Our coaching structure was developed to provide rehearsal and understanding of skills in many contexts so that individuals could see the conditions under which a new procedure applied. Rehearsal of skills is presented every time the student asks for help. The first hint he receives is a structured rehearsal of what he has done on that problem, so he can reflect on his plans and actions. Further, each time additional help is requested he gets a trace of the hints already received. Thus, the actions, outcomes, conclusions and options for a plan or action can all be reviewed during the solution process,

Understanding is fostered in many contexts. Each node can give hints about the conditions under which certain things would be true. For example, *"These data signals should cause relays K3 and K2 in the A1A1A13 card to set. One way to determine if the A1A1A13 card is operating properly is to do an ohms check on the relays being set by the stimuli code select thumbwheel switches. You could ohm out the lower contacts of K3 (between Pins 1 & 3) and the lower contacts of K2 (between Pins 55 & 57) to determine if the relays are being set."* The content of the hints are specific to the work environment, realistic and detailed.

The role of experts is essential to the development of efficent scripts and procedures for fault isolation. The history taking procedure is an efficent script for electronics troubleshooters. In history taking the technician examines what devices are active for the current test step and looks at the previous tests that have passed. Thus he builds a set of hypotheses as to where the fault might lie, and also can disconfirm hypotheses, by comparing the test step that is currently failing with steps that have passed. Hypotheses are generated by determining what in the current test differs from other tests. For instance, if a measurement device was used to measure voltage in test 1 and test 1 passed, and used again in test 2 in the exact same way but test 2 failed, one can infer that this measurement device is not the problem . However, if the type of measurement changed from one test to another, the measurement device for the second test would be a suspect device since it is a new addition to circuitry already demonstrated to be sound. This is one example of an expert procedure tutored by Sherlock.

Processes of Acquisition. Understanding the processes of acquisition is necessary for sequencing of instruction to maximize learning. We approached this by identifying the knowledge that differentiated skilled and less-skilled performers. A major finding in our cognitive task analysis research was that skilled airmen did not do better than less skilled technicans on discrete knowledge components such as basic electronics knowledge or basic operations, such as knowing where to put meter probes on a circuit board to get a particular measurement. Instead, expert troubleshooters were better primarily at orchestrating all of the cognitive competencies necessary for fault location. Our results support Estes' (1976) observation regarding performance on intelligence measures: that it is not enough to identify cognitive processes, but rather it is how these processes become organized for efficient performance that distinguishes higher levels of proficiency. Given our findings, we decided that the processes of acquiring troubleshooting skills had to be taught in the context of a

troubleshooting problem so that individuals would learn how to orchestrate their skills rather than just be able to perform such skills when the cognitive load was low. We recognize that the practice of some component tasks until they become automated through practice may enable more effort to be directed to higher level tasks (Schneider, 1984; Perfetti & Lesgold, 1979). However, component processes that work well separately may not work well together. If a task demands orchestration of component skills, then assessment must be able to differentiate skill inefficiencies from skill absences. Sherlock assesses component skills in the context of assessing performance levels for individual curriculum issues. The trainee who does poorly on one skill but well on another will receive coaching based on the learned state for each particular issue. Different levels of hint specificity are provided based on the learned state for particular skills. Thus we sharpen those skills that need to be sharpened in the context of the problem space.

Understanding the knowledge a trainee brings to the learning environment makes the task of instruction simpler since instruction then consists of coherence building (Resnick, 1985) or restructuring the individual's current knowledge to a higher level of comprehension. The learning process consists of connecting new information to existing knowledge. Learning takes place on the basis of existing mental models and theories held by students, which can either enhance or retard learning (Glaser, 1985). Glaser states that with proper instruction students can test, evaluate, and modify their current theories on the basis of new information.

Sherlock reinforces prior knowledge and builds up mental models of performance through specific coaching. Both a general mental model of the troubleshooting process and a specific mental model of a test are taught. Understanding the basic troubleshooting scenario includes such things as collecting the correct resources, understanding the current test, analyzing device history, and tracing schematics. The mental model of a test, for example, consists of a stimulus input to the unit to be tested, a circuit-adjusting load, and a measurement of output. Other mental models deal with automatic configuration of equipment to carry out a test. Sherlock's coaching structure addresses these mental models in each problem. Sherlock reinforces these mental models by providing hints that teach individuals self-regulatory or metacognitive skills. Specific tutoring on device and system understanding link the student's problem solving ability to his prior knowledge.

Self-regulation is taught by providing recapitulations of the student's problem solving trace when a trainee asks for help. These hints encourage him to reflect on his decision-making process by illustrating his prior hypotheses. If a trainee does something inefficient the tutor tells him why his actions are redundant given his prior problem solving activities. Sherlock teaches the trainee to constrain the problem space and efficiently test hypotheses. When a trainee retests a unit that was already ruled out, Sherlock will say something like, *"you have already eliminated this device as a problem area by swapping the card with a good card, why don't you try something else?"*

Sherlock confronts the student to consider his hypotheses, make conclusions, and then choose alternative plans and actions based on test outcomes. It tutors device and system understanding by linking test procedures to actual problem solving performance. If, for instance,a trainee tried to perform actions that specifically were contrary to the test steps being used, then Sherlock would say something like the following, *"Testing this card is not necessary since it is not being used in this test. When you enter Measurement Code Select 12 and press Enter, you should be activating TPA12. Perhaps you should check your schematics again."*

Thus, student actions are linked to system knowledge by pointing to the information in the technical resources that highlight the relevant device knowledge. A theory of competence and an understanding of the processes of acquisition were used to develop Sherlock. Our theory of intervention links these two theories in the form of the coached practice environment. However, there are other things that a theory of intervention must consider.

Theory of Intervention. Research must test the effects of particular forms of teaching and instructional materials on particular students. Sherlock has the advantage of being able to monitor the trainee's progress through instructional materials and provide adaptive forms of instruction based on individualized performance.

Individual differences in aptitudes and learning styles play an important role in learning from instruction, and no one form of instruction fits every individual's needs (Cronbach & Snow, 1977). Some individuals are much better with spatial-visual instruction whereas others derive more meaning from verbal-textual representations. Sherlock minimizes the effects of learning differences by providing multiple medium presentations.

Just as different modes of instruction must be considered so should the focus of instruction. There is some controversy as to whether or not educational curricula should focus on general thinking skills, specific skills or both types of knowledge at the same time. Some would argue that knowledge competencies in a domain and the ability to think about that domain develop hand in hand (Bransford, Sherwood, Vye & Reiser, in press). Studying a domain requires a split focus, where one focus is on the material itself and another focus is on checking to see that the appropriate mental operations are produced to facilitate learning and problem solving (Locke, 1975; Simon, 1980).

Even though teaching both specific knowledge skills and general problem solving skills simultaneously is difficult and demanding, there is evidence that this approach to instruction has long term benefits. An excellent example can be given for teaching reading comprehension. Palinscar & Brown (1984) provided a learning environment where inquiry was encouraged and students were encouraged to be consciously aware of the strategies they used in the context of reading. Strategies were taught with procedures for where, when, and how to use them. Comprehension monitoring strategies were taught by encouraging students to check their inferences and organize their knowledge. This split focus of instruction was successful: students made significant improvements in the quality of their summaries and questions as well as overall gains on criterion tests of reading comprehension. Furthermore, these skills were demonstrated to transfer to novel tasks.

General and specific skills were tutored by Sherlock by providing opportunities for interrogation or inquiry, by teaching troubleshooting procedures and the conditions under which such procedures were useful, and finally by an emphasis on self regulation. Inquiry was supported in two ways: (a) students could ask for assistance anywhere in the problem space, and; (b) Sherlock would sometimes respond by asking questions itself.

Were the Instructional Strategies Effective? Given the instructional strategies that underlie Sherlock we can ask whether theory driven design of a coached based practice environment improves instruction. The answer in our case is yes. A pre/post test comparison of electronics troubleshooting performance of a treatment and control group was examined and the results indicated that the tutored group did significantly better than a control group on post test performance ($X2$ (1, $N = 77$) = 3.67, $p < .05$, tutor group mean problems solved = 39, control group mean problems solved = 21). In addition the tutor group was shown to significantly increase their number of expert moves ($F(1,27) = 28.85$, $p < .000$, tutor group mean expert moves in post test = 19.33, control group mean = 9.06) and decrease bad moves ($F(1,27) = 7.54$, $p < .01$, tutor group mean bad moves in post test = 1.89, control group mean = 4.19). In an independent analysis, Gott (1988) found that the tutored subjects' performance rose by the equivalent of 47 months on-the-job experience as a result of the Sherlock experience. Tutoring individuals to test their hypotheses, to reflect on the results of their measurements, to understand the procedures and conditions under which such procedures work, resulted in positive performance gains. The treatment group increased their proportion of expert like tests providing evidence that tutoring a mental model of a test improved performance.

However, there were several instructional strategies used in our coached environment and we must discover which of these strategies were reponsible for Sherlock's success.

References

Bransford, J., Sherwood, R., Vye, N., & Rieser, J. (in press). Teaching thinking and problem soving: Research foundations. *American Psychologist.*

Cronbach, L. J. & Snow, R. E. (1977). *Aptitudes and instructional methods: A handbook for research on interactions.* N.Y.: Irvington.

Estes, W. K. (1976). Intelligence and cognitive psychology. In L. B. Resnick (Ed.) *The nature of intelligence.* Hillsdale, N.J.: Erlbaum Associates.

Glaser, R. (1985). *The integration of instruction and testing.* Paper presented at the Educational Testing Service, Invitational Conference on the Redesign of Testing for the 21st Century. New York City.

Gott, S. (1988). Personal Communication.

Leinhardt, G. (1986). Expertise in mathematics teaching. *Educational Leadership, 43,* 28-33.

Lesgold, A., Lajoie, S., Bunzo, M., & Eggan, G. (1988). *A coached practice environment for an electronics troubleshooting job.* Presented at the J. S. McDonnell Foundation Conference on CAI and ITS, March, Carnegie Mellon University, Pittsburgh PA.

Lesgold, A., Lajoie, S., Eastman, R., Eggan, G., Gitomer, D., Glaser, R., Greenberg, L., Logan, D., Magone, M., Weiner, A., Wolf, R., Yengo, L. (1986). *Cognitive Task Analysis to Enhance Technical Skills Training and Assessment.* (Technical Report), Pittsburgh: University of Pittsburgh, Learning Research and Development Center.

Lesgold, A., Lajoie, S., Logan, D., & Eggan, G. (In Preparation). Cognitive task analysis approaches to testing. In N. Fredericksen, R. Glaser, A. Lesgold, & M. Shafto (Eds.), *Diagnostic monitoring skill and knowledge acquistion.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Locke, E. Q., (1975). *A guide to effective study.* NY: Springer.

Palinscar, A. S., & Brown, A. L. (1984). Reciprocal teaching of comprehension- fostering and monitoring activities. *Cognition and Instruction, 1,* 117-175.

Perfetti, C., & Lesgold, A. (1979). Coding and comprehension in skilled reading. In L. B. Resnick & P. Weaver (Eds.) *Theory and practice of early reading.* Hillsdale, NJ: Erlbaum.

Resnick, L. (1985). *Cognitive science and instruction.* Technical Report No. 13, Learning Research and Development Center, University of Pittsburgh.

Schneider, W. (1984). Practice, attention, and the processing system. *Behavioral and Brain Science, 7,* 80-81.

Simon, H. A. (1980). Problem solving and education. In D. T. Tuma, & F. Reif (Eds.), *Problem solving and education: Issues in teaching and research.* Hillsdale NJ: Erlbaum.

Creatures of Habit
A Computational System to Enhance and Illuminate the Development of Scientific Thinking *

Roy Pea
New York University

Michael Eisenberg†
Franklyn Turbak
Massachusetts Institute of Technology

Abstract

Creatures of Habit is a computer-based microworld designed to engage middle-to-high school students in the process of scientific inquiry. The system depicts a universe of interacting programmable "creatures" whose individual behavior is guided by simple rules that may model naive psychology, physical laws, chemical affinities, and other domains. Students can create or revise creature rules and explore the resulting (and often surprising) emergent behaviors within "artificial ecosystems"; or they may employ predesigned ecosystems in undertaking more structured problem-solving activities. Our objective is for students to use these ecosystem simulations as an enjoyable introduction to a variety of scientific domains, especially the area of *dynamical systems*, a field of science where experiments with such simulations often leads theory. The system encourages a wide range of reasoning and learning central to scientific methodology — pattern observation, hypothesis formation, experimentation, data collection and analysis, and deduction. We describe the rationale behind the system, discuss some sample activities, and outline the system's potential both as a learning environment and as a research laboratory for empiricial studies of scientific thought. Finally, we briefly describe the present state of our prototype Creatures of Habit system.

KEY WORDS: microworlds; dynamical systems; scientific reasoning

Introduction

"Doing science" involves learning to use complex techniques and skills — making observations, noticing interesting patterns, forming hypotheses and theories, making conjectures, and designing and running experiments. If introducing students to scientific content were sufficient, education would be hard enough; studies show children bring ill-formed models of scientific domains such as mechanics, electricity, chemistry, and biology to school (Driver, 1985; Osborne, 1985). Students need to learn not only a body of structured beliefs, but how to participate in the processes of science, which are intimately related to content. Students must understand that scientific ideas are motivated and supported by theory, experiment, and argumentation — not authority. Beyond the content and process of science, students should also enjoy doing science. They often view science as a mysterious, unapproachable culture in which they observe rather than participate. A truly effective science education should dispel this image by giving students opportunities for designing and refining inquiries.

We describe a computational system under development — "Creatures of Habit" (henceforth, Creatures) — designed to address these science education issues by providing rich, exploratory, and enjoyable scientific experiences for the middle-to-high school years. Creatures is a microworld of programmable interacting "creatures" whose behaviors are based on rules. In exploring this microworld, students can be introduced to important content in various scientific domains, notably in the area of *dynamical systems*; moreover, the program allows for a wide range of activities central to doing science — such as conjecture, theory formation, experimentation, deduction, and communication of results. Perhaps most important, Creatures is intended to lead students toward original and creative work — to have them participate in science as fledgling researchers driven by curiosity, rather than onlookers motivated by assignment.

Besides its utility for science education, we see roles for Creatures in empirical studies of scientific thought. Because it affords activities engaging reasoning skills

*This research and development project was supported by a grant from the Spencer Foundation awarded to Roy Pea, under a project entitled "Intelligent Tools for Education."

†Bell Labs Ph.D. Scholar

such as analysis, synthesis, and evaluation (see below), it may be used to examine how these different skills develop. The system can thus provide a uniform environment in which to design specific instructional and experimental tasks.

In the remainder of this paper, we first motivate the design of Creatures by discussing the subject of dynamical systems; we then describe the system[1], outline some learning activities that might be undertaken with it, sketch how it reflects key learning goals for science education, and propose a design for an experiment in which Creatures may be used to illuminate the development of scientific thought.

Dynamical Systems: Analytic Science, Synthetic Science, and "New Wave" Science

Since Creatures is designed to provide introduction to concepts and methods involved in the study of complex dynamical systems (Thompson & Stewart, 1986), we briefly describe the growing importance of such science. Certainly there has been an explosion of interest in this subject within the scientific community (Gleick, 1987); and this phenomenon has ramifications both for scientific methodology and for science education.

Historically, scientific thought has been characterized as either "analytic" or "synthetic" (Oldroyd, 1986). In this classic formulation, the analytic method is observing phenomena and seeking laws to account for them; the synthetic method involves confirming the validity of laws via prediction and experimentation. The advent of scientific computing has added a new texture to this division. In studying a system, a scientist may account for its behavior by constructing an abstract model that can be realized as a program; that model may then be simulated by running the program as a test of its applicability. Thus, there is an *analytic* side to computer modeling (observing real-world phenomena and designing model systems that might illuminate them); and a *synthetic* side (changing parameters in the model system to match observations or see new phenomena). Hut and Sussman (1987) describe this approach as "analysis by synthesis"; while Farmer and Packard (1986) call it "new wave science" — a methodology "characterized by attempts at synthesis rather than reduction, cutting across conventional disciplinary boundaries.... New 'effects' are discovered through a combination of insight and serendipity, and more often than not experiment leads theory....Simulations are frequently used to develop qualitative insight, often by studying highly simplified models which are nonetheless complicated enough to possess universal properties found in more complicated systems."

Developing computer simulations as experimental systems has led to a blossoming scientific literature beyond cognitive science. Simulations are employed to discover robust properties of intergalactic collisions, heat flow in solids, kinetics of chemical reaction mechanisms, evolutionary adaptation, and many other dynamical systems. These simulations enable the study of increasingly complex phenomena; but they also tax skill in developing formal models, and present students with new phenomena to understand and methods to use. Techniques of developing, studying, and documenting computer models have become an integral element of scientific method. In this context we see Creatures as having particular potential value, since it provides an engaging "introduction to complexity" and an environment in which to learn-by-doing these skills of "analysis by synthesis." As we will show below, students can construct systems of creatures that exemplify concepts such as stability, oscillations, and bifurcations. Because these systems may be of their own devising, the concepts become personalized in a way that canned demonstrations could not.

Creatures of Habit: The Basic Elements

Creatures is an environment in which students can explore "artificial ecosystems" composed of interacting programmed entities called "creatures." Inspired by Braitenberg's (1984) *Vehicles*, these creatures behave and interact on the computer screen according to sets of simple rules. The rules might be chosen to reflect physical laws, naive psychology, chemical affinities, and so on. A simple example of a group of creatures and how they might interact illustrates the system's components.

Consider the scenario illustrated by the sequence of "snapshots" in Figures 1 to 5.[2] Here we see three different creature types (distinguished by geometric shape) with the following behavioral rules:

1. *Squares* are attracted to squares, but indifferent to triangles and circles;
2. *Circles* are repelled by triangles and circles, but attracted to squares;
3. *Triangles* are attracted to circles, but indifferent to squares and triangles.

Figure 1 shows an initial "ecosystem" with five creatures, and Figures 2 – 5 demonstrate how this ecosystem evolves over time according to these rules (note that creatures leave a visible path). Even in this simple case we see many types of behavior: mutual attraction (between squares), mutual repulsion (between circles), and predator-prey relationships (the triangle "chases" the circle that is "running away" from it). Note how the interactions can be parsed into "episodes" in a naive model of animal behavior:

- The squares like each other and begin to move together. The circles, interested in the squares but oblivious to each other (and the triangle), tag along. The triangle, spying a tasty circle, starts sneaking up on it;

[1] Creatures exists in prototype form on a Hewlett-Packard Series 300 Model 320.

[2] These figures were generated by our prototype Creatures system.

Figure 1: Initial Configuration

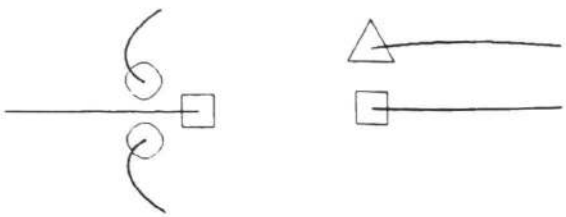


Figure 2

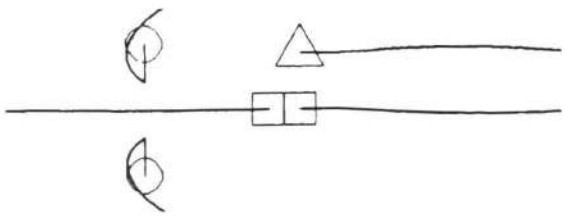


Figure 3

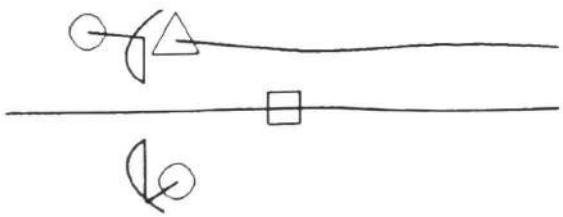


Figure 4
(Note that the two squares overlap.)

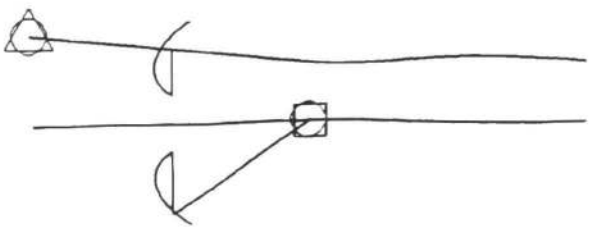


Figure 5: Final (Equilibrium) Configuration

- After noticing one another, the frightened circles flee in opposite directions;
- The triangle chases the topmost circle while the other circle, regaining its composure, heads back toward the squares.
- The creatures reach a final "equilibrium" configuration when the triangle "catches" the topmost circle and the other circle joins the two squares.

This simple scenario indicates how even a small number of creatures governed by straightforward rules can lead to interesting, varied behavior. Under slightly different situations, it is possible to observe more complex behavior and "emergent phenomena"[3]. Even this scenario suggests interesting questions to explore, e.g., how do initial positions of the creatures influence the episodic nature of the interaction or the creature "fates"? How would adding one creature to the start state affect the interaction? Given these species, is it possible to design a configuration that exhibits stable oscillations?

We now present a fuller exposition of the basic elements of the Creatures of Habit system:

Creature Morphology

Creatures are small mobile fantasy creatures that "live" on the screen. Creatures come in many "species," identified by a set of discrete characteristics. Above, creatures were distinguished only by shape, but more animal-like creatures are conceivable (a species might be identified as purple, with bobbling eyes, pointy ears, squiggly tail). Species may also be endowed with less visible properties such as mass or birth rate.

Species Rules

Creatures interact with each other and their environment by obeying a small set of species-specific rules linking perception to action. Typically, species rules indicate which properties are "attractive" or "repellent" to a species. Once a species rule-set is defined, every member of that species uses those rules to govern its behavior.

Ecosystem Rules

The complete dynamics of a population of creatures is determined by a higher-level set of ecosystem rules which specify how the creatures employ the notions of "attraction" and "repulsion" when moving. Ecosystem rules are perhaps best illustrated by example; the particular examples that we provide should be read merely as indicating a space of possibilities. Variations on these rule-sets are of course possible, and other rule-sets are computationally viable.

Ideal Gas Rule-Set. In this simple model, all creatures are regarded as indifferent to one another (i.e. properties that are attractive/repulsive to a given creature's species do not affect its motion). The ecosystem rule-set does include, however, a default collision rule

[3] For instance, an entire cluster of creatures might move as one unit due to the internal patterns of attraction and repulsion between the cluster's individuals.

specifying that collisions between creatures are elastic and momentum-conserving[4]. Note that this rule-set implicitly assumes that mass will be included among the species properties. Creatures here are regarded as inanimate billiard-ball-like objects, each with a (species-specific) mass.

Inverse-Square Force Rule-Set. This rule-set elaborates the last one. Creatures are particles upon which inverse-squared forces act. To determine the individual force of creature *C2* on creature *C1*, we sum the attractions (+) and repulsions (-) that *C1* has for *C2*'s properties. The resulting number divided by the square of the distance between *C1* and *C2* is the magnitude of this individual force; the direction between *C1* and *C2* determines the direction of the individual force. The total force on creature *C1* is the vector sum of the individual forces from all other creatures. Dividing this total force by creature *C2*'s mass determines the acceleration, or velocity change, of *C1*. (Again we assume creatures have mass as a species-specific property — and we might add some other properties, such as "charge," as well.)

Distance-Dependent "Animal Attraction" Rule-Set. This rule-set embodies a simple "naive psychological" model for creature interactions; it was the ecosystem rule-set implicitly used in the scenario depicted in the figures. To determine how much creature *C1* would "like/not-like" creature *C2* if it were one unit away, look at *C1*'s species rules and subtract the number of repellent properties exhibited by *C2* from the number of attracting properties. To determine how much *C1* *actually* likes *C2*, divide the previous result by the distance; this inverse distance dependence models a situation in which creatures pay more attention to closer objects. Thus, if creature *C1* finds three properties of creature *C2* attractive, and one repellent, then the unit-distance strength of *C1*'s attraction for *C2* is 2; if *C1* is 10 units of distance from *C2* at that moment, then *C1*'s current attraction toward *C2* has a strength of 0.2. To determine how creature *C1* should move, the system finds the creature about which *C1* feels most strongly (like/not-like); creature *C1* should then move at a constant (species-specific) speed towards that creature (if liked) or away (if not).

Encounter-Dependent Birth and Death Rates. In this rule-set based on population biology, we assume that creatures have species-specific properties of "birth and death rates." Creature movement might be governed by the rule-set above, and in addition new species members may be born or die. The birth rules might depend upon encounter rates between members of the same species. The birth rule for species *A* might be the following: on any time step, an *A*-creature has a 2% chance of giving birth to another *A*-creature, except when the previous time step resulted in contact with an *A*-creature, in which case there is a 50% chance of giving birth. The death rate of a species might be defined similarly (e.g., on any particular time step, an *A*-creature has a 3% chance of dying, except when the previous step resulted in contact with a "predator" species B, with a 60% chance of dying).

[4] It is also possible to collide with the boundaries of the screen; these collisions should be treated as elastic collisions with a stationary wall.

Interface to the System; Additional Tools

We have described the basic elements of Creatures but thus far ignored the interface. In this section, we describe several aspects of the interface design for Creatures. Many issues must be resolved to make Creatures both accessible to first-time users and extensible for use by expert scientists. We aim to first provide some beginning ecosystem rule-sets for easy access to Creatures; given a choice of ecosystems, the user may edit the properties and rules for individual species with iconic menus. The basic interface should also provide a fair amount of control over simulations: initial configurations may be specified by directly moving creatures to their desired starting positions, and individual runs may be paused, single-stepped, replayed, and stored.

The description above does not address the needs of more experienced users who wish to create or edit the ecosystem rule-sets. The range of ecosystem rule-sets may be constrained so that they might be customized via a menu interface; but more likely, a more elaborate interface to ecosystem rule-sets (verging on a special-purpose programming language) will be required.

We are working on other special-purpose tools for the Creatures system for measuring, modifying, and experimenting with configurations and rules. For example, we aim to provide multiple screen windows so the student can make two runs at once, comparing results from two systems by varying only one parameter — say the initial position of a creature. Experimentation will also be supported by "annotation windows" where students enter information obtained from the results of previous runs; e.g. to examine a sequence of runs and make deductions about the species-rules of individual creatures, structured tables help a student codify the information obtained and highlight the information needed for a solution. The system will include graphical measuring devices that enable the student to take precise measurements of distance, angle, and time; in our experience with the Creatures prototype, these variables have proven crucial in describing important features of simulations.

Sample Activities

Exploratory Activities

The most fundamental activity we imagine is exploration. Here, students are provided with sets of creatures in an ecosystem, and observe how the creatures behave and interact. Since both the species rule-sets and the ecosystem rule-sets are accessible, students can investigate the relationship between the behavior they observe and the rule-sets governing that behavior. And because the system facilitates change at a wide variety of levels (from the number and positioning of the creatures to the species and ecosystem rule-sets), students can readily explore the effects of "perturbing" a given scenario in diverse ways. Even inexperienced students could examine how the qualitative behavior

of a configuration depended on the initial position of one of the creatures; more advanced students could explore the effects of "tweaking" species rule-sets; yet more advanced students could work at the ecosystem rule-set level.

Even within this purely exploratory frame, a number of important scientific ideas are introduced. To highlight, it quickly becomes apparent in working with Creatures that *complex systems may arise even from very simple rules*. Often these systems exhibit emergent properties — stability, oscillations, irreversibility — that resist explanation in terms of low-level rules. Moreover, by playing around with these systems, students can develop important skills that are usually slighted by classroom science: making conjectures, searching for patterns in data, and generating qualitative or statistical descriptions. We emphasize that although exploration is a low-threshold activity, it is not merely an introductory one; rather, it is the essential point of the design of Creatures. Indeed, although Creatures is targeted for students in middle-to-high school, issues raised by "mere exploration" could evolve into highly challenging problems at or beyond the college level.

Problem-Solving Activities

Although Creatures lends itself to exploration, it supports more structured "problem-solving" activities as well. We have used the prototype to play a "Mystery Creatures" game where users are provided with a set of creatures whose species rules are unknown and must be determined from their behaviors. The goal of the game is to design experimental runs and thereby deduce the rule-sets of the creatures. Sometimes, one well-chosen experiment can illuminate the behavior of a number of mystery creatures all at once; in other situations, the user must design a sequence of experiments. A related game, Invisible Creatures, involves several visible creatures (whose rule-sets are accessible) that interact with an "invisible" creature. In this case, the goal is to determine the identity and location of the invisible creature based on the observed actions of the visible ones. These kinds of activities stress skills of experimentation and logical deduction.

Design Activities

Many activities supported by Creatures have an important design focus. Here the emphasis is not on *why* a given system evolves as it does; rather, the goal is to *construct* a system exemplifying some desired behavior. For example, given a set of creatures, one might ask: "Is there an initial configuration for these creatures such that all the creatures will collide at once?" Design can take place at many levels, from constructing configurations of creatures to "creating new worlds" with new ecosystem rules.

A Hypothetical Scenario

The following scenario shows how a student might use Creatures. We use the same species and ecosystem rules as in the scenario shown in the figures earlier:

▷ The student begins with a screen on which a dozen creatures — some squares, triangles, and circles — move in various directions, creating intricate patterns. The student tries similar runs using the same numbers of creatures, but with different initial configurations, just to get a feeling for the kinds of phenomena that might occur.

▷ The student decides to figure out what the species rules for each of the three creature types might be. She runs a sequence of experiments in which two creatures of the same type are placed at a slight distance from each other in the screen center. She observes that when triangles are used, they stay where they began; squares move toward each other; and circles move away from each other. She concludes that triangle creatures are indifferent to other triangles, while squares are attracted to other squares and circles are mutually repelled.

▷ When she places a square and two circles in a particular initial configuration, the two circles move toward the square until, at a later time, they drift apart; then, shortly after, they move toward the square again. The student hypothesizes that circles find the square attractive, but when they are too close to each other, their mutual repulsion becomes stronger than their attraction for the square. She replays the earlier run, this time occasionally pausing the simulation and measuring distances and headings of the creatures to test her conjecture.

▷ Replaying the previous run, the student notices the square remained stationary throughout. She concludes that squares are indifferent to circles. In fact, she recalls that in her earlier simulations, whenever there was only one square on the screen, that square did not move at all regardless of how many circles and triangles were about; so she concludes that squares must be indifferent to triangles, too.

▷ During her experiments with circles, the student remembers noticing that the two creatures ended up standing still in opposite halves of the screen.[5] She decides to investigate: Does any initial configuration of any number of circles always end up in a "stationary" state; and if so, do the final states reveal some pattern?

Clearly, the scenario above could be extended in different ways. For example, the species rules could involve features besides shape; or the creatures could move at varying speeds; or the student might choose to examine whether any "stationary configuration" remains stationary if one other creature is added to the screen; or the student could design a new creature that moves away from squares, and see what happens when it is placed in a crowd of (mutually attracting) squares. From this very simple beginning, many projects — some touching on very sophisticated questions — may arise.

Creatures as Part of A Science Curriculum

A perennial tension in science education exists between teaching content knowledge and scientific

[5] It should perhaps be mentioned that, for this particular scenario, we assume that the screen on which the creatures move "wraps around" in both the x- and y-directions.

method. The flaw in overemphasizing content is that it tends to lead to rote learning, with little sense of how scientists work to develop theories. This approach typically fails to recognize students' pre-existing conceptions of scientific domains (diSessa, 1987); students are tacitly encouraged to memorize laws and results without ever engaging their own reasoning powers to see how they were arrived at, or could be tested. In contrast is "discovery learning", which emphasizes experimentation and the scientific process. The difficulty with this approach is twofold. First, the notion of "discovery" is often realized in practice by canned experiments; the "discovery," whose results are known beforehand, is superficial. In this kind of setting, students spend much of their time trying to find the "right answer," rather than what the experiment means. Second, when discovery learning is realized most sincerely, by simply letting the student explore on his or her own, the teacher is often left with questions about what the student's explorations were, and what was learned (Hawkins, 1987).

We feel that Creatures can serve as a medium for guided, exploratory discovery-type learning (cf. White and Horwitz, 1987). The worlds students can explore with Creatures are brand-new; there are few "right answers," and real discoveries can be made. But while making these discoveries, students can obtain a structured introduction to the role of conjecture, measurement, and experimentation in the scientific process. Activities such as the "Mystery Creatures" game place an emphasis on logical reasoning; these puzzle-like activities can provide the benefits of occasional settings in which a right answer does in fact exist.

An Experimental Design for Studying the Development of Scientific Thought

We have stressed the role of Creatures in science education. Here we illustrate how it may also be used for studying the development of scientific thought, proposing an experimental design we aim to undertake using the next iteration of the system. Consider the following three tasks, corresponding to three common modes of scientific thought, that a student might be asked to perform using a particular ecosystem and starting creature configuration:

1. The student watches the system run (i.e. creatures move on the screen), and is asked to infer the species rules of each creature. This *analytic* task involves working backward from observed behavior to underlying rules.
2. The student is shown the species rules for the creatures, then asked to predict the result of running the system using the startup configuration. This is a *synthetic* task of predicting behavior from a known set of deterministic rules.
3. The student is provided the set of species rules for the creatures, *and* shown the results of running the system using the startup configuration; his task would simply be to explain events observed in the scenario in terms of the species rules. This is an *evaluative*, or perhaps *explanatory*, task.

Now the very same ecosystem and starting configuration could be used for each of these three tasks, as given to three different groups of students; but the kinds of descriptions generated for each of the three tasks — and indeed, the features of a particular scenario attended to — might vary widely. By obtaining think-aloud protocols from these different subject groups, we could begin to develop a coherent picture of how the several modes of scientific thinking differ; specifically, we could begin to distinguish these modes according to the sorts of phenomena to which they typically apply.

As a possibility, we might ask whether the phenomena used to deduce rules in the "analytic" tasks are the same phenomena for which explanations are offered in the "evaluative" task. In other words, are the phenomena that people find salient for the deduction process the same as those that people find representative in the explanatory process? For example, in looking for underlying rules, subjects might tend to focus on events in which creatures reverse direction (e.g., a situation in which creature *A*, in moving toward an "attractive" creature *B*, finds itself too near a "repellent" creature *C* and turns around). In contrast, subjects given the task of explaining a given scenario might insist on a chronologically faithful narrative; i.e., they might pay as much attention to explaining creature A's initial direction as they do to explaining its change of direction.

Or we might try to characterize the kinds of "episodic groupings" that different task groups assign to the given scenarios: e.g., are there certain tasks in which scenarios are typically viewed as "punctuated" by changes in direction? Are there certain tasks in which the final states (e.g., the achievement of equilibrium) are attributed more importance? Under what circumstances do people reason at the "system level," talking about behaviors of larger groups of creatures, rather than at the "atomic level" of individual creatures' histories?

Yet another issue involves the possibility of an order effect between different activity modes. For example, in a different study we could ask subjects who have just predicted a scenario in the "synthetic" task to watch the actual scenario and perform the "explanatory" task; their protocols could be compared with those generated by subjects given the explanatory task alone. We could then begin to characterize how a subject's previous predictions affect the kinds of explanations generated for a particular scenario.

Finally, it is worth mentioning the issue of noticing "creature indifference" in the context of this experimental design. In our attempts at watching scenarios and deducing the rules underlying them, we found that a great deal of information is conveyed by noting which creatures are indifferent to which others. Indifference between creatures seems to be a subtler notion that attraction or repulsion; when watching a scenario, there is a tendency to attribute a creature's movements solely to attraction or repulsion without

recognizing that the *absence of motion* toward or away from some other creature is also informative. This tendency is reminiscent of the disinclination to use negative information in scientific reasoning (Mynatt *et. al.*, 1977; Wason, 1977). Thus another question addressable within this experimental design is whether any of the three tasks is more likely to lead subjects to confront the issue of creature indifference.

Conclusion: Present Status, Future Directions

We now have two prototype Creatures of Habit systems: one written in Scheme (a Lisp dialect), one in C. Both run on a Hewlett-Packard Series 300 Model 320 computer. The Scheme version contains facilities for developing and editing ecosystem and species rulesets, and includes features for maintaining histories of individual runs. But it is slow and at the moment unsuitable for running systems with more than 4-5 creatures. The C program is less elaborate or interactive, but its running speed is two orders of magnitude faster than the Scheme program; using the C prototype, simulations employing 16 creatures have been run at a satisfactory speed. We continue work with both systems, using Scheme as a medium for developing new tools and trying out ecosystems in the small, and the C program to run simulations quickly and experiment with properties of larger creature-worlds.

Much of our current design effort is developing a suitable interface to the system, and delineating a "starter set" of sample ecosystems appropriate for middle-to-high school students. A second theme noted above is using the system as a laboratory tool for conducting experiments in the development of scientific thought. It should also be possible to follow up with work in student modeling and intelligent "coaching" efforts to promote the various inquiry skills in science we have outlined. Ultimately, we hope that Creatures of Habit will provide an environment in which students can acquire the concepts, the methodological techniques, and — too frequently neglected — the intellectual curiosity of the working scientist.

References

Braitenberg, V. (1984). *Vehicles*. Cambridge: MIT Press

diSessa, A. (1987). Towards an intuitive epistemology of physics. UC Berkeley Cognitive Science Report #48.

Driver, R.*et. al.*, eds. (1985). *Children's Ideas in Science*. Philadelphia: Open University Press.

Farmer, J. and Packard, N. (1986). Evolution, Games, and Learning: Models for Adaptation in Machines and Nature. In Farmer *et. al.* (Eds.), *Evolution, Games, and Learning*, pp. vii-xii. New York: Elsevier.

Gleick, J. (1987). *Chaos*. New York: Viking Press.

Hawkins, J. (1987). The Interpretation of Logo in Practice. In R. Pea and K. Sheingold (Eds.), *Mirrors of Minds*, pp. 3-34. Norwood: Ablex.

Hut, P. and Sussman, G. (1987). Advanced Computing for Science. *Scientific American*, 255:10, 144-153.

Mynatt, C.R., Doherty, M.E., and Tweney, R.D. (1977). Confirmational Bias in a Simulated Research Environment: An Experimental Study of Scientific Inference. *Quarterly J. Exp. Psy.*, 29, 85-95.

Oldroyd, D. (1986). *The Arch of Knowledge*. London: Methuen & Co.

Osborne, R.*et. al.* (1985). *Learning in Science*. Auckland: Heinemann Publishers.

Thompson, J. & Stewart, H. (1986). *Nonlinear Dynamics and Chaos*. Chichester: John Wiley and Sons.

Wason, P.C. (1977). "On the failure to eliminate hypotheses..." — a second look. In P.N. Johnson-Laird and P.C. Wason (Eds.), *Thinking*, pp. 307-314. New York: Cambridge University Press.

White, B. and Horwitz, P. (1987). Thinkertools: Enabling Children to Understand Physical Laws. BBN Report No. 6470. Bolt Beranek and Newman Inc., Cambridge, Massachusetts.

PROPOSITIONAL ATTITUDES, COMMONSENSE REASONING, AND METAPHOR

John A. Barnden

Computing Research Laboratory
New Mexico State University

THE CLAIM TO BE ARGUED

People have highly metaphorical ways of regarding each others' minds, as is well known and will be exemplified later. This means that in practice people will often adopt metaphorical views of each others' propositional attitudes (their states of belief, hope, expectation, intention, desire, and so on). This has been ignored in detailed technical studies of how to represent propositional attitudes in AI systems. Yet, this paper will argue that it is absolutely crucial to pay attention to such views in designing propositional attitude representation schemes.

The paper briefly argues this claim about metaphor, which I have made recently elsewhere [Barnden 1988], but also places it in the context of the following more general claim:

> *In the design of representation schemes for beliefs and other propositional attitudes, it is essential to pay attention to issues of commonsense, plausible reasoning (including metaphorical reasoning as a special case), as opposed to the strict deductive reasoning that is traditionally considered.''*

We shall refer to this as The Claim. There are, certainly, some well-recognized connections between propositional attitudes and commonsense, plausible reasoning, and these links have already affected attitude-representation design to some extent. However, the paper will press for a new effect on such design. Also, the *strongest* previous effects on attitude-representation design have come not from commonsense reasoning but from deductive reasoning (see next section). This accounts for the contrast to deductive reasoning in The Claim.

The plan of the paper is as follows. This section will briefly review some existing links between propositional attitudes and commonsense reasoning. The next section will unify important aspects of the traditional interplay of deductive reasoning and propositional attitudes under a certain banner (the ''inference-lifting question schema''), but will argue that this interplay is just a special and somewhat artificial special case of a more general interplay that allows for commonsense reasoning also. Then, three sections will lead up to showing the importance in this interplay of *metaphorical* reasoning, because of people's metaphorical views of attitudes. We will see that attitudes and commonsense reasoning are more intricately intertwined than is commonly realized — to the extent that commonsense plausible reasoning must be given a bigger role in the design of genuinely useful propositional-attitude representation techniques than it has had in the past. This role is at last sketched in the penultimate section.

As for recognized connections between attitudes and commonsense reasoning, the following ones are among the most important. A central aspect of discourse and speech act theory is listeners' plausible conclusions about the intentions, beliefs, etc. of speakers. [See e.g. Allen 1983, Grosz & Sidner 1986, Sperber & Wilson 1986]. Automated tutoring systems should ideally make plausible inferences about the beliefs, expectations and intentions of users. Another connection is shown in some research that has been done on resource-limited belief reasoning [e.g. Konolige 1983; see also Fagin & Halpern 1987]. Commonsense reasoning is involved in choosing between different types of reading of propositional attitude reports, e.g. between ''de-re'' and ''de-dicto'' readings [Wiebe & Rapaport 1986]. There has recently been research on mapping between plausible-reasoning logics and belief logics [Konolige 1987]. Maida (1986) has worked on introspection-based, plausible ascription of belief to other agents. Issues of plausible, default-based ascription of beliefs have been addressed by Ballim (1987), Wilks & Ballim (1987), and Barnden & Ballim (1988) argue that the issues have a (not well-recognized) effect on attitude-representation design.

COMMONSENSE REASONING AND ATTITUDES: INTRODUCTION

The Claim concerns the issue of *how inference requirements affect detailed representational design* for propositional attitude purposes. Most relevant work within AI and philosophy has been on

(a) respecting the referential opacity of propositional-attitude contexts;

(b) preventing "logically omniscient" inference [see e.g. Levesque 1984, Fagin & Halpern 1987];

(c) while, in some research, allowing simple types of logical combination or analysis of beliefs, as in inferring to and/or fro between "Z believes that R and S" on the one hand and "Z believes that R" and "Z believes that S" on the other.

Issues (a) to (c) are mainly concerned with the following question schema:

IL: The Inference-Lifting Question-Schema

GIVEN that the statement Q follows, in sense F, from some statements P,

DOES the statement "Z believes (hopes, etc.) that Q" follow, in sense G, from the statements "Z believes (hopes, etc.) that P"?

The term "statement" is used in a loose, general sense in this paper, covering both natural language sentences and formal representational expressions.

The following will assume that, in the schema, A's propositional attitude is one of belief (by far the most common case discussed). In one important version of issue (a), there are two P statements, e.g. "Jim's wife is clever" and "Mary is Jim's wife"; Q is derived from the first of these by co-referential substitution, giving "Mary is clever"; and senses F and G are both deductive consequence. The answer to the question is "no" when "Jim's wife" in the statement "Z believes that Jim's wife is clever" is interpreted in a de-dicto way. Note also that the referential opacity issue is also taken to cover the case when Mary is Jim's wife but agent Z is *not* presumed to believe this.

In issue (b), the P statements are a statement R and a statement to the effect that R implies Q, and senses F and G are again deductive consequence. Logical omniscience is commonly taken to include other slightly different patterns of deductive inference about agent Z [see e.g. Fagin & Halpern 1987], but these are less germane to this paper.

Thus we see that issues (a), (b) and, obviously, (c) are primarily concerned with F and G being *definite* (i.e. deductive) inference, as opposed to *plausible, commonsense* inference, even though the issues are indeed genuinely concerned with commonsense inference in the indirect or suppressed sense that the proscriptive thrust of (a) and (b), and the permissive thrust of (c), are inspired by commonsense observations *by the theorist* on what it is reasonable to infer about ordinary agents.

However, just as most inference of true practical interest in AI and cognitive science generally is of a commonsensical and merely plausible nature (based on defaults, abduction, metaphor, and so on), the instances of IL that are of *primary* interest are those in which sense F and/or sense G brings in plausible, commonsense reasoning. One such commonsense instance of IL has A's attitude being belief, Q being "Peter can fly", the P being the statements "Usually birds can fly" and "Peter is a bird", and F being the sense of default reasoning. G is also likely to be some plausible-reasoning sense, although notice that the degree of plausibility in the inference to "Z believes that Q" from the "Z believes that P" has no necessary, simple connection with the degree of plausibility that would obtain if the system were to infer Q from the P. For example, the system may happen to know that John almost always does in fact makes (at least tentative) inferences about the flying abitliy of particular birds, so that the inference to "John believes that Q" could have a much higher plausibility than the system's potential inference to Q from the P.

The confidence inequality can be the other way around: Q might even follow *definitely* from the P, as far as the system is concerned, and yet the system may only have limited confidence in the

conclusion that Z believes Q. Thus even the traditional issue of IL instances where F is a definite-reasoning sense, *G should be taken to be a plausible-reasoning sense.*

I have not explored any implications that IL instances such as the one about birds might have for attitude-representation design. We proceed now to build a bridge towards IL instances that do have such implications.

COMMONSENSE INSTANCES OF THE IL SCHEMA: "EXPLICATIONS"

Consider the attitude report: *Professor Z believes that X's theory is threatened.*

An AI system interpreting this sentence may well have to *explicate* the notion of threat involved here, in order to adequately link the information conveyed by the sentence with other information. For instance, suppose that the system believes that Professor Z is usually correct in her beliefs about whether there are experiments providing evidence against given theories. Suppose also that one reasonable explication of the notion of a theory being threatened (out of perhaps many possible explications available to the system) is that *there is an experiment that provides evidence against the theory.* Then, if the system considers it reasonable to impute this explicated view of theory-threat to Z, Z can use the attitude report to make the plausible inference that

Professor Z believes that there is an experiment providing evidence against X's theory.

The system can go on to use its belief about Z's reliability on such matters to make the plausible inference that there is indeed an experiment providing evidence against X's theory.

We have here an instance of the IL schema in which P is the statement that X's theory is threatened, and Q is the statement that there is some experiment providing evidence against X's theory. It is clear that G is some sense of *plausible inference.* As for sense F, we could suppose that Q follows from P in the sense of being a plausible *analysis* of the meaning of P. Or, we could suppose that Q follows from P in the sense of being an ordinary plausible *inference* from P. This sort of distinction actually makes little difference to the concerns of this paper.

In the example we started with the natural language sentence displayed above, but there is no implication that the conclusion (Prof. Z believes that there is some experiment ...) is also a natural language sentence. Rather, we may take the system to be inferring from the given sentence to some internal representational expression for the conclusion. We are here unconventionally subsuming the natural language understanding process under inference. This subsumption is mainly for the sake of simplicity of discussion, and it makes little ultimate difference to the issues of concern in this paper.

It is potentially a complex matter for the system to devise a reasonable explication of a vague notion, like that of a theory being threatened, embedded in a belief. In our example, the explication can depend on the academic fields of X and Z (and these fields may differ), default information about how people in Z's field view theories in X's field and their relationship to experiments, special information about Z's view of these matters, and so on. Also, there can be explicit hints about specific explications, as in "Professor Z believes that X's theory is threatened by Y's more economical theory". Here an experiment-based explication that might otherwise have been used would be discarded in favor of an explication in terms of opposing theories. The complexities in the choice of reasonable explications are beyond the scope of this paper but are the subject of active research by the author.

METAPHORICAL INSTANCES OF THE IL SCHEMA

So far we have ignored the possible metaphorical nature of the threat notion. Let us now assume that the notion is metaphorical, and that the vehicle of the metaphor is a WAR domain [see e.g. Lakoff & Johnson 1980] in which people can threaten, battle with, and conquer each other, and so on.

It is then important for the AI system to be able to relate the example sentence of the last section to statements such as the following, which stay within the same metaphor:

Prof. Z believes that Y intends to attack X's theory,

given that this sentence is interpreted to mean that the attacking aspect of the WAR metaphor *is in fact being used mentally by Z.* (This reading is analogous to de-dicto interpretation of referential phrases in belief contexts.) If one of the reasonable explications of "threatened" that is available to the system is to the effect that someone is intending to attack the threatened thing, then the system could come to the plausible conclusion that the threat to X's theory believed by Z is in fact just Y's intention (according to Z) to attack it. Notice that this conclusion rests not only on the plausibility of the indicated explication of theory-threat within the indicated metaphor, but also on the plausibility of ascribing this metaphorical view to Z.

Very similar issues arise with an example sentence like

General Z believes that terrorism is creeping across the globe.

Suppose the system has been told that "The belief that terrorism is slowly and stealthily moving across the globe is excessively pessimistic". The system should be able to come to the plausible conclusion that Z is being excessively pessimistic. What is at issue here is, again, commonsense inference *within* a particular metaphor. This type of inference serves for sense F in the IL schema. Sense G is some form of plausible inference. P is the statement that terrorism is creeping across the globe, and Q the statement that terrorism is slowly and stealthily moving across the globe. It seems implausible and uneconomical to suggest that the AI system should connect the two metaphorical statements about terrorism by going through some literal explications. Probably no adequate ones are available anyway.

Thus, the situations with which this section began, in which a word like "threatened" indicates a metaphor, and in which a metaphorical explication of the word's import should be constructed, are just a special case of a more general metaphor-based inferential phenomenon subsumed by the IL schema. However, it is the special case that leads us into the crux of the argument, as follows.

METAPHORICAL EXPLICATIONS OF ATTITUDES

The notions within attitude contexts that need to be given possibly-metaphorical explications *include propositional attitudes themselves.* Consider the sentence

Z hopes that X will come to realize that his theory is faulty.

We will see that the propositional-attitude notion of realizing may have to be treated analogously to the notion of being threatened — and, in particular, that it is likely that the notion of realizing will have to be given some *metaphorical* explication that is suggested by discourse context. The argument is a condensation of the more detailed discussion in Barnden (1988). During the argument it is important to bear in mind that what is at issue is the explications and metaphors that *the agent Z* can be reasonably taken to have in mind for the notion of realizing.

Commonsense, metaphorical views of mind that people entertain have received some close attention [see e.g. Lakoff & Johnson 1980, Larsen 1987, Reddy 1979, Tomlinson 1986]. Now, suppose the above sentence is succeeded by the following one, which brings in the metaphor of X's mind as a battleground in which forces (ideas, habits, etc.) engage in struggles:-

But Z is afraid that the idea of the theory being faulty is having an uphill struggle against X's habit of wishful thinking.

Assume that it is reasonable to take Z herself (not just the speaker) to be thinking in terms of the metaphor. Then the most commonsensically reasonable and coherent way to approach the two sentences is to explicate, in terms of that metaphor, the realization by X that is hoped for by Z in the first sentence. That is, the system should, say, take Z *to hope that the idea of X's theory being faulty will gain dominance in the battleground of X's mind.*

It might be argued that instead of *introducing* a metaphor into the interpretation of the first sentence, the system should seek to *eliminate* the metaphorical aspect of the second sentence. This argument, however, faces the well-known difficulty of eliminating metaphors for mental states/processes [see e.g. Fainsilber & Ortony 1987]. Also, the mind-as-battleground metaphor might be continued and further elaborated in succeeding sentences, such as:

Z has had much experience of conflicts of this sort in her colleagues.

In sum, what is needed is a system that can explicitly cast an agent, such as Z in the example, as using a particular metaphor, or mix of metaphors, chosen from a range of possible metaphors for propositional attitudes (the attitude of realizing, in the example). Notice also that the chosen metaphor(s) may differ from default ones the system might be tempted to use for a realizing that does not appear within another attitude. The system might, for instance, use by default the metaphor of realizing as being a matter of solidifying an idea into concrete form.

THE EFFECT ON ATTITUDE REPRESENTATION

When the AI system meets the sentence "Z hopes that X realizes that his theory is faulty" and interprets it, the interpretation might well use some standard, atomic way of representing the state of realizing, as for instance in the modal-logic formula

hopes(Z, realizes(X, faulty(theory-of(X)))).

Thus, there is a limited sense in which the representation of attitudes is not affected by the considerations of the previous section. However, these considerations show that in order for the internal, formal rendering of the sentence to do useful work in relation to later sentences (or, of course, earlier ones), it will typically have to be replaced, or at least supplemented, by a rendering in which X's realization is explicated in appropriate metaphorical terms. As a simplified illustration, the new rendering could be:

hopes(Z, will-dominate(idea-of(X, faulty(theory-of(X))), battleground-in(mind-of(X)))).

The thesis is that *this* is the sort of representation (of what it is that Z reportedly hopes for) that will be heuristically useful.

Notice that the effect on representation that has been claimed comes down to an effect on the sort of predicate symbols (for instance) that are used, on the assumed ontology underlying the representation, and on the degree of elaborateness of attitude representations. The arguments have not displayed a reason for thinking that traditional *styles* of representation — e.g. modal, quotational, concept-based, situation-based[1] — are inadequate. However, my previous work on attitude representation [Barnden 1983, 1986a,b, 1987a,b] shows that the use of quotational, concept-based and situation-based styles has a natural tendency to lead to a *commitment to a particular explication* of an attitude that is nested inside and attitude of an agent Z, where, moreover, the explication is typically a *commonsensically implausible* one to impute to Z — whereas, of course, we have argued for the use of any of a *variety* of *commonsensically plausible* explications to impute to Z. Commitments to particular explications can be entirely avoided, but only at a significant cost of complexity in the representational syntax (see Barnden (1987a,b) for the quotational case). It is, on the other hand, relatively easy to replace the explications the styles tend to enforce by commonsensically plausible ones, using formulae analogous to the second one displayed in the previous section. Thus, the present paper's considerations do impinge upon the broad style of representation, to the extent of recommending that if the mentioned styles of representation should only be used with caution and in a particular way.

[1] See e.g.: for modal style: Chellas 1980, Halpern & Moses 1985; for quotational style: Haas 1986, Perlis 1985, Quine 1981; for concept-based style: McCarthy 1979, Creary 1979, Creary & Pollard 1985; for situation-based style: Barwise & Perry 1983, Hobbs 1985.

The modal-logic style was not addressed in the previous paragraph, but there are considerations mitigating against its usefulness in practical contexts, including: the often-noted inability to quantify over beliefs and treat them as individuals; and the inability to deal with (metaphorical) belief explications that are explicitly signalled, as in "Z believes that some experiment, of whose identity she has only a hazy idea, is a threat to X's theory" [Barnden 1987a,b]. This sentence suggests that a belief is composed of ideas which can be hazy.

CONCLUSION

We saw that a consideration of the metaphorical explications of propositional attitudes nested within other attitudes has a fundamental effect on how we should approach the task of designing a system that can represent attitudes so as to reason about them adequately in practical situations.

The issue of metaphorical attitude-explications is within the intersection of two more general issues: that of metaphors within attitude contexts, and that of commonsensical explications of notions (e.g. "threatened") within attitude contexts. These two issues are in turn about special commonsense cases of the Inference-Lifting schema IL, and illuminate the much more general point that the instances of schema IL that are of primary interest are those where senses F and/or G bring in *commonsense, plausible* reasoning, as opposed to deductive reasoning as is traditional.

A detailed representational approach based on the ideas of this paper is being developed from the quotational-cum-concept-based scheme described in Barnden (1987a,b).

ACKNOWLEDGMENTS

I am grateful for encouragement, stimulation, and constructive criticism from Afzal Ballim, Dan Fass, Sylvia Candelaria del Ram, and Yorick Wilks.

REFERENCES

Allen, J. (1983). Recognizing intentions fron natural language utterances. In M. Brady & R.C. Berwick (Eds), *Computational models of discourse*, Cambridge, Mass.: MIT Press.

Ballim, A. (1987). The subjective asciption of belief to agents. In J. Hallam & C. Mellish (Eds), *Advances in artificial intelligence*. Chichester, UK: Ellis Horwood.

Barnden, J. A. (1983). Intensions as such: an outline. *Procs. 8th Int. Joint Conf. on Artificial Intelligence*, Karlsruhe, W. Germany.

Barnden, J.A. (1986a). Imputations and explications: representational problems in treatments of propositional attitudes. *Cognitive Science, 10* (3), 319-364.

Barnden, J.A. (1986b). A viewpoint distinction in the representation of propositional attitudes. In *Proceedings of the 5th Nat. Conf. on Artificial Intelligence (AAAI86)*, Philadelphia, Penn.

Barnden, J. A. (1987a). Interpreting propositional attitude reports: towards greater freedom and control. In B. du Boulay, D. Hogg & L. Steels (Eds), *Advances in artificial intelligence - II*, Elsevier (North-Holland): Amsterdam.

Barnden, J.A. (1987b). Avoiding some unwarranted entailments among nested attitude reports. *Memoranda in Computer and Cognitive Science*, No. MCCS-87-113, Computing Research Laboratory, New Mexico State University.

Barnden, J.A. (1988). Belief, metaphorically speaking. Submitted to 1988 European Conf. on Artificial Intelligence (ECAI-88).

Barnden, J.A. & Ballim, A. (1988). The relevance of belief environments. Submitted to 7th National Conf. on Artificial Intelligence (AAAI-88).

Barwise, J. & Perry, J. (1983). *Situations and attitudes*. Cambridge, Mass.: MIT Press.

Chellas, B.F. (1980). *Modal logic*. Cambridge University Press.

Creary, L. G. (1979). Propositional attitudes: Fregean representation and simulative reasoning. *Procs. 6th. Int. Joint Conf. on Artificial Intelligence*, Tokyo.

Creary, L.G. & Pollard, C.J. (1985). A computational semantics for natural language. *Procs. 23rd Ann. Meeting of the Association for Computational Linguistics*, Univ. of Chicago.

Fagin, R. & Halpern, Y.J. (1987). Belief, awareness, and limited reasoning. *Artificial Intelligence, 34* (1), 39-76.

Fainsilber, L. & Ortony, A. (1987). Metaphorical uses of language in the expression of emotions. *Metaphor and Symbolic Activity, 2* (4), 239-250.

Grosz, B.J. & Sidner, C.L. (1986). Attention, intentions, and the structure of discourse. *Computational Linguistics, 12* (3), 175-204.

Haas, A.R. (1986). A syntactic theory of belief and action. *Artificial Intelligence, 28*, 245-292.

Halpern, J. Y. (ed.) (1986). *Theoretical aspects of reasoning about knowledge: proceedings of the 1986 conference*. Los Altos, CA: Morgan Kaufmann.

Halpern, J.Y. & Moses, Y.O. (1985). A guide to the modal logics of knowledge and belief: preliminary draft. In *Procs. 9th. Int. Joint Conf. on Artificial Intelligence*, Los Angeles, 1985.

Hobbs, J.R. (1985). Ontological promiscuity. *Procs. 23rd Ann. Meeting of the Association for Computational Linguistics*, Univ. of Chicago.

Konolige, K. (1983). A deductive model of belief. *Procs. 8th Int. Joint Conf. on Artificial Intelligence*, Karlsruhe, W. Germany.

Konolige, K. (1987). On the relation between default theories and autoepistemic logic. *Procs. 8th Int. Joint Conf. on Artificial Intelligence*, Karlsruhe, W. Germany.

Lakoff, G. & Johnson, M. (1980). *Metaphors we live by*. Chicago: U. Chicago Press.

Larsen, S. F. (1987). Remembering and the archaeology metaphor. *Metaphor and Symbolic Activity, 2* (3), 187-199.

Levesque, H. J. (1984). A logic of implicit and explicit belief. *Procs. Natl. Conf. on Artificial Intelligence*, Univ. of Texas at Austin.

Maida, A. S. (1986). Introspection and reasoning about the beliefs of other agents. *Procs. 8th Conference of the Cognitive Science Society*.

McCarthy, J. (1979). First order theories of individual concepts and propositions. In J. E. Hayes, D. Michie & L. I. Mikulich (Eds.), *Machine Intelligence 9*. Chichester: Ellis Horwood.

Perlis, D. (1985). Languages with self-reference I: Foundations. *Artificial Intelligence, 25*, 301-322.

Quine, W.V.O. (1981). Intensions revisited. In W.V. Quine, *Theories and things*. Cambridge, Mass: Harvard U. Press.

Reddy, M. J. (1979). The conduit metaphor — a case of frame conflict in our language about language. In A. Ortony (Ed.), *Metaphor and Thought*, Cambridge, UK: Cambridge University Press.

Sperber, D. & Wilson, D. (1986). *Relevance: communication and cognition*. Oxford: Basil Blackwell.

Tomlinson, B. (1986). Cooking, mining, gardening, hunting: metaphorical stories writers tell about their composing processes. *Metaphor and Symbolic Activity, 1* (1), 57-79.

Wiebe, J.M. & Rapaport, W.J. (1986). Representing *de-re* and *de-dicto* belief reports in discourse and narrative. *Procs. of the IEEE, 74* (10), 1405-1413.

Wilks, Y. & Ballim, A. (1987). Multiple agents and the heuristic ascription of belief. *Procs. 10th Int. Joint Conf. on Artificial Intelligence*, Milan, Italy.

A Process-Oriented, Intensional Model of Knowledge and Belief

by

Robert F. Hadley

School of Computing Science
Simon Fraser University

1. INTRODUCTION

Within the cognitively related disciplines (including AI) there are three dominant approaches towards the modeling of epistemic states. For convenience we call these the logistic, syntactic, and intensional approaches. The *logistic* approach seeks to discover a *logic* of belief (knowledge) which would enable one to derive, *a-priori*, the totality of things an agent believes (or knows), given the formulas explicitly written in the agent's belief base.[1] The *syntactic* approach (as Levesque [1984] dubs it) identifies an agent's beliefs with just those formulas explicitly written in the agent's belief base -- the agent has no other beliefs. By contrast, the *intensional* approach identifies beliefs with abstractions, e.g., with *thought contents* or propositions. On this account the same belief may be expressed by means of a limited variety of distinct syntactic objects. As will emerge in section 1.3, our present concerns lie with the intensional model, and with how intensions are related to the *processes* of inference, belief acquisition, and belief retraction. However, we now briefly describe the logistic and syntactic approaches, and consider how these differ from the present work.

1.1 The Logistic Approach

Those who adopt the logistic approach are concerned with the discovery of *inference rules*, such that an agent's explicit beliefs are *closed under* the application of these rules. Typically, the epistemic logics proposed do not require an agent's beliefs to be closed under *all* valid inference rules (full logical omniscience is not required), but closure under *certain* inference rules is still required. Thus, these logics typically require that agents each possess an infinity of beliefs, or at least believe an infinity of distinct formulas. (cf. [Levesque, 1984; Halpern & Moses, 1985; Lakemeyer, 1987]). Some have questioned the motivation for introducing logics which represent agents as believing an infinity of formulas (e.g., [Fagin & Halpern, 1985]), but the underlying rationale may be that such logics employ a non-standard sense of belief, which yet has application in the context of finite agents. For example, the sense of belief appropriate to [Levesque, 1984] *might* be such that "*p* is believed" means approximately that one could *eventually come to believe p* by a tractable computational process. However, whether or not one thinks that epistemic logics (of the "infinite belief set" variety) require us to adopt a non-standard sense of belief, such logics usually are not concerned with certain important *process-related* aspects of belief. For example, with limited exceptions[2] such logics do not address the fact that an agent acquires and *changes* beliefs over time, or that these processes often involve inference, testing beliefs for consistency, retraction, etc. Doubtless, most epistemic logics were never intended to address those aspects of belief, but for certain applications one desires a *formal model* of how these cognitive processes relate to the belief states of the agent. An agent *X* who knows what another agent *Y* believed an hour ago may wish to form a reasonable conjecture about what *Y* believes *now*. If so, *X* needs to know what *processes* must be assumed to exist in order to arrive *even* at a reasonable conjecture about *Y*'s present beliefs. Of course, some aspects of belief change are addressed by truth maintenance systems (e.g., [de Kleer, 1986; Doyle, 1979]), but TMSs are concerned with *procedures* for manipulating belief sets, rather than with formal models which interrelate the concepts of belief, knowledge, inference, retraction, etc. By contrast, we are here concerned with formalizations of epistemic states, and the processes which affect those states.

1.2 The Syntactic Approach

[1]Hereafter, "belief" shall be taken to include knowledge, unless the context requires that the distinction be explicit.

[2]Konolige, [1985] presents an epistemic logic which does address the process of belief *introspection*. However, Konolige's system is not intended to address most other process-related aspects of belief.

Now, apart from epistemic *logics* which posit an infinity of beliefs, certain *syntactically-oriented* belief models do address the fact that beliefs can be acquired and retracted by *processes* (cf. [Haas, 1985]). In general, those who adopt the syntactic approach identify beliefs with particular syntactic objects, so that, for example, an agent might believe the formula $p \vee q$ but not $q \vee p$, or believe "Tadpoles swim" but not "Pollywogs swim". Now, many reject the syntactic approach entirely, on the grounds that it is too fine-grained. The syntactic approach arguably fails to account for the fact that *some* syntactically distinct formulas seem to express precisely the same thought, and therefore ought to have the same belief status (cf. [Levesque, 1984] ; [Hadley, 1986]).

Fagin and Halpern [1985] present what might be viewed as a *liberalized* syntactic belief model. The Fagin-Halpern (F-H) model identifies an agent's beliefs with a subset of formulas of which the agent is aware. This particular subset (which satisfies special conditions) *may* be taken to include formulas the agent will only *come to be* aware of at some future time. In this respect the F-H model is less restrictive than those which restrict an agent's beliefs to formulas presently written in the agent's belief base. However, apart from the awareness requirement, the F-H model is not concerned with the empirical preconditions of belief (e.g., that a sophisticated robot might derive a formula from existing beliefs, but not "commit to" the formula until a "consistency check" with other beliefs is complete).

There is no reason to suppose that the F-H model was ever intended to deal with the empirical preconditions of belief, but an agent who wishes to reason (or conjecture) about what a given robot *currently* believes will require a model of belief which makes explicit many of these preconditions. Such a model will need to recognize the fact that agents revise and retract beliefs, and that they sometimes derive formulas which they *never* accept, even though the derived formula is indeed a valid consequence of the agent's existing beliefs.

1.3 The Intensional Approach

The intensional approach is founded upon the premise, widespread among philosophers of language, that belief and knowledge must be understood as a relation between an agent and an abstract *intension* (proposition or thought). Recent work in epistemic modeling, which adopts an intensional stance, includes that of Wilks and Ballim [1987], and Maida [1986]. Wilks and Ballim are concerned primarily with the *generation* of nested beliefs (especially atypical beliefs) and Maida concentrates upon epistemic problems which arise between agents whose reasoning strategies are unknown to each other. Nevertheless, these authors share our assumption that agents believe a particular symbolic representation (sentence, formula) insofar as they believe the intension *expressed* by that representation.

The thesis that a declarative sentence expresses an abstract sense or intension, whose structure is a *function* both of the sentence's syntax and of the meaning of its parts, is a view which dates back at least to Frege. Modern philosophical descendents of Frege, including Montague [1970] and Lewis [1976], have considerably refined and modified Frege's view, but Frege's underlying idea that the sense (intension) of a sentence is determined by a compositional process, involving both syntax and meaning functions, persists. The precise definition of 'intension' varies from theory to theory, but a sentence's intension is usually taken to be a *function* which, given a possible world as argument, returns that truth value of the sentence in that possible world.

One notorious difficulty with intensional semantics is that when intensions are merely construed as functions in the set-theoretic sense, then every tautology, ϕ, turns out to have the same intension (or meaning), since each valid ϕ must be assigned the value 'true' in each possible world. (Recall that ϕ's intension is a function from possible worlds to truth values). Thus, when intensions are regarded *merely* as functions, intensions are too coarse-grained to serve as meanings or "thought contents". In an attempt to circumvent this difficulty, Lewis [1976] defines sentence meaning as a function both of a sentence's *parse tree* and its intension. In doing so, he quickly disposes of the problem concerning tautologies. However, many believe Lewis's modification results in a theory of meaning which is excessively fine-grained, since under his modification virtually *every* syntactically distinct pair of sentences (e.g., $p \vee q$ *vs.* $q \vee p$) is assigned a different meaning. In [Hadley, 1988] a simple modification of Lewis's view is proposed to remove this difficulty. The modification creates a degree of granularity for intensions which allows us to say that *structurally* non-isomorphic formulas (including tautologies) differ in intension, but formulas which are both logically equivalent and structurally isomorphic in a graph-theoretic sense, have the same intension or meaning. For example, on this account $p \vee q$ and $q \vee p$ have the same intension since they are logically equivalent and their parse trees are isomorphic.

Many other *compositional* theories of sentence meaning exist, of course, and each has its own degree of granularity. Cresswell [1985] proposes a theory of sentence meaning with an attractive degree of granularity, as does Gawron [1986]. For present purposes, we shall remain neutral among these differing theories. We shall simply identify intensions with sentence meanings, and the reader may select the theory of meaning which suits his/her purpose. What is relevant for our purposes is that each of the aforementioned theories proposes a *canonical* system for representing meanings or intensions, such that *all sentences which are synonymous* (according to the semantic theory) *are mapped onto a single canonical representation.* Moreover, each of the semantic theories includes a *principled* set of rules for translating (or compiling) an expression into its canonical meaning representation. Questions about synonymy may be answered by comparing the canonical representations into which the respective expressions are "compiled" by the translation rules. Questions whether two distinct expressions can represent the same belief will in turn be decided by ascertaining whether the distinct expressions are synonymous.[3]

1.4 The Proposed Model

In what follows an axiomatic system is presented which aims to formalize many of the interrelationships among the epistemic concepts, and related notions such as retracting, questioning, or committing to a belief. The model is designed to capture human-like epistemic concepts, and is intended to be of use to those who seek *formal* models of epistemic *processes.* Unlike many who adopt the *logistic* approach, our overriding concern is not with finding a tractable model, but with providing a conceptual map which is sensitive to the empirical contingencies of belief and knowledge.[4] Moreover, because our focus is primarily upon *cognitive* fidelity, it is hoped that the axiomatic structure will make vivid just how many assumptions one must make to conjecture that an agent has some *actual* belief at a given time.

The epistemic model to be described is axiomatized in first-order logic. In spite of this, the model permits the representation of *nested* beliefs, and permits agents to reason about the beliefs of themselves and others. Self-reference is possible within the language of the model; thus, the potential for paradoxes of self-reference arises. A method (due to Kripke [1975]) for eliminating the contradictions which self-reference can engender is described in section 3.

In section 2 the axioms of the formal model are presented. The axioms modify and generalize the *intensional* model of belief proposed in [Hadley, 1988] (which included neither *knowledge*, the representation of nested epistemic propositions, nor self-referential belief statements). As will become apparent, the model proposed here (hereafter 'NIM' for "nested intensional model") requires an agent to know or believe *only* those formulas which are intensionally identical with formulas explicitly present in the agent's belief base. Thus, NIM does not entail logical omniscience. Moreover, NIM does not require us to suppose that agents believe every formula they have validly *derived* from other beliefs.

2. AN INTENSIONAL EPISTEMIC MODEL

The axiom set described below (section 2.1) permits agents to reason about their own epistemic states and those of others. Thus, the language of our model includes nested (embedded) epistemic formulas. However, the language should also permit us to say that different agents believe different formulas at different times. For this reason, epistemic assertions should involve at least three separate parameters. The representation of assertions involving multiple parameters is straightforward in first-order logic (FOL), and for this reason (among others) FOL is chosen. However, the representation of *nested* epistemic propositions in FOL is less straightforward.

Perlis (1985) describes a method, involving quotation, unquotation, and concatenation of expressions, which permits the representation of nested propositions in an extension of FOL. However, this method can be unwieldy for levels above the first level of embedding. For this reason, we here adopt the simple expedient of treating the epistemic operators (K and B) as skolem functions. For example, we will suppose that the function $K(i, f, t)$ simply returns the *proposition* that agent i knows formula f at time t. When we wish to assert this proposition, we simply attach the predicate 'true', as in $true(\ K(i, f, t)\ )$. (We assume lower case for predicates and variables, and upper case for

[3]The reader is referred to [Hadley, 1988] for a detailed discussion of the compilation process and the individuation of intensions.

[4]There is no reason to suppose that all relevant aspects of belief and knowledge can be expressed in tractable formal systems. We know, for example, that humans engage in certain forms of reasoning which are intractable. For more on this, see Perlis [1985].

functors). We do not attempt to define the *truth* predicate within our axiom set, but assume it to be interpreted from without (cf. section 3). Moreover, we need not *always* explicitly display the truth predicate; rather we stipulate that any functor written in bold face is preceded by an implicit truth predicate. So, to say that (for all agents, formulas, and times) if it is true that a formula is known to be known, then that formula is true, we may write:

$$\mathbf{K}(i, K(j, f, t1), t2) \rightarrow true(f).$$

(Unless otherwise indicated, assume variables are universally quantified with widest possible scoping). Note that the foregoing assumptions approximate the Fregian view that when a formula occurs within an epistemic context, it denotes its *sense*, and otherwise, it denotes its truth value.

2.1 Axioms of the Model

Before presenting the axioms of NIM, it may prove helpful to make explicit a few more background assumptions. In the following we assume that every agent possesses a belief base (BB), which contains a finite collection of explicit formulas which the agent regards as true. Any axiom which the agent accepts must be represented in the BB by at least one formula. Synonymous (intensionally equivalent but syntactically distinct) formulations of such axioms need not be present. Beliefs which are *caused* in the agent via processes not normally considered inferential (e.g., via sense perception) will be regarded, for present purposes, as axioms acquired at various times. In addition, the BB may contain formulas which the agent has explicitly inferred and "accepted". We assume that all formulas in a BB belong to an artificial language, but we make no assumptions about the particular language and inference rules which the agent X employs. We do *not* assume that X's axioms form a consistent set, or that X never makes a fallacious inference.

We further assume that agents *may* employ strategies for maintaining consistency, and may, as a consequence, retract previously held beliefs. The retraction process may, or may not be ideally rational. Finally, we do not assume that an agent enters a sentence into his BB as soon as it is inferred. There is a sense of 'infer' according to which one believes anything one has inferred, but here we use 'infer' to mean merely 'derive according to accepted rules (which may include heuristics)'. Agents may treat newly inferred formulas as "candidates" for belief, and enter an examination or "questioning" phase during which the candidate is tested for consistency with all or some of the agent's other beliefs.

We now present the axioms of NIM. As will be apparent, NIM contains an unusual number of predicates whose interpretation is underconstrained by the formal structure. This is due, we believe, to the fact that it deals with *aspects* of belief which are simply ignored in most *axiomatic* treatments. Our contention is that one simply cannot form reasonable conjectures about an agent's belief states over time, unless these additional aspects are explicitly considered. For clarity, the intended interpretations of each axiom and predicate are discussed as the axioms are introduced.

The expression $\mathbf{B}(i, f, t)$, which occurs in most of the axioms, may be interpreted as *it is true that i believes f at time t*. The predicate *in-bb(i, f, t)* may be rendered *formula f is explicitly present in i's BB at time t*.

1. $in\text{-}bb(i, f, t) \rightarrow \mathbf{B}(i, f, t)$

Axiom 1 tells us that all formulas explicitly present in *i*'s BB at time *t* are among *i*'s beliefs at *t*. In axioms 2 and 3, below, the intended interpretation of *sense(f,s)* is: *s is the intension of formula f*. The variable *s* ranges over the class of intensions. Thus, axiom 2 states that an agent who believes a formula *f* believes all formulas synonymous with *f*. Axiom 3 makes an analogous statement about knowledge. Recall that, for artificial languages at least, the synonymy of formulas may be ascertained by compiling the formulas into their canonical intensional representations, and checking the results for identity.

2. $[sense(f_1, s) \wedge sense(f_2, s)] \rightarrow [\mathbf{B}(i, f_1, t) \equiv \mathbf{B}(i, f_2, t)]$

3. $[sense(f_1, s) \wedge sense(f_2, s)] \rightarrow [\mathbf{K}(i, f_1, t) \equiv \mathbf{K}(i, f_2, t)]$

The following axiom expresses the widely accepted view that whoever knows *f* must at least believe *f*, and *f* must be true.

4. $\mathbf{K}(i,f,t) \rightarrow \mathbf{B}(i,f,t) \wedge true(f)$

Note that, if we assume $true(K(Mary, K(John, p, t1), t2))$ (which we abbreviate as $\mathbf{K}(Mary, K(John, p, t1), t2)$), then we may use axiom 4, together with conjunctive simplification, to derive $true(K(John, p, t1))$. This in turn enables us to derive $true(p)$, by analogous reasoning. Axiom 5 simply states that an agent who retracts a formula ceases to believe the formula. This, together with axiom 4, entails that the formula would not be known. We may use axiom 2 to show that the agent would also cease to believe all synonymous formulas. Axioms 1 and 5 together entail that a retracted formula ceases to be in the belief base.

5. $retracts(i,f,t) \rightarrow \neg\mathbf{B}(i,f,t)$

In the following axiom, *questioning(i,f,t)* may be interpreted as *i begins at time t to investigate whether formula f is consistent with other beliefs*. The underlying idea is that, if *i* has begun to question whether *f* should be retained, then *i* is not presently committed to *f*'s truth. We assume that those automated reasoning systems, which employ belief revision strategies, enter a phase in which the consistency of a set of formulas is investigated before any retractions are made. A formula which is being questioned may be one which has been singled out for special investigation. However, we do not insist that 'retracts' and 'questioning' be interpreted just as suggested here, and no attempt is made to formalize these notions. If a given system lacks a questioning mechanism, or even a retraction mechanism, then the 'questioning' predicate (or 'retracts') may be assigned the value 'false', which would render the presence of these predicates innocuous.

6. $questioning(i,f,t) \rightarrow \neg\mathbf{B}(i,f,t)$

The intended interpretation of axiom 7 is that any formula which has just been inferred by an agent, from a subset of that agent's beliefs, is to be regarded as a candidate for belief by that agent. In the following, *just-inferred(i,f,s,t)* may be read *i inferred f from set s at time t*. *Mem(w,s)* may be read as *w is a member of set s*, and *cand(f,i,t)* is rendered as *f is a candidate for i at t*.

7. $[just\text{–}inferred(i,f,s,t) \wedge \forall w \{ mem(w,s) \rightarrow in\text{–}bb(i,w,t) \}] \rightarrow cand(f,i,t)$

Axiom 8 tells us that, under normal circumstances, beliefs persist. More specifically, the axiom tells us that if *i* believes a formula at t_1, then *i* also believes the formula at all later times, unless *i* retracts or questions the formula at some intervening time. Since the interpretation of 'retracts' is left to the user's discretion, 'retracts' may be taken to include some form of spontaneous forgetting or other memory loss. Nothing here forces us to restrict retraction to deliberate actions. In 8, *later*(t_1,t_3) may be read as t_3 is later than t_1, and *inc-between*(t_1,t_2,t_3) may be read t_2 *lies inclusively between* t_1 *and* t_3.

8. $[\mathbf{B}(i,f,t_1) \wedge later(t_1,t_3) \wedge$
$\neg\exists t_2 \{ inc\text{–}between(t_1,t_2,t_3) \wedge [retracts(i,f,t_2) \vee questioning(i,f,t_2)] \}] \rightarrow \mathbf{B}(i,f,t_3)$

The preceding eight axioms are intended to be *descriptive* of our concept of belief. They relate the concept of belief to cognate concepts in a manner intended to be useful to agents who reason about the beliefs of an artificial agent. The following axiom, by contrast, is *prescriptive*. It is intended to convey the conditions under which a candidate formula (in the sense introduced by axiom 7) *should* become an active belief. The term '*successful*(i,f,t_2)' should be taken to mean that at time t_2 formula *f* has just been successfully scrutinized by agent *i*'s questioning mechanism, and has been found acceptable. The term '*should-be-inserted*(i,f,t_2)' may be read as, formula *f* should be added to *i*'s BB at time t_2. Thus axiom 9 states that a formula which is a candidate for *i* at t_1, and which has successfully passed the questioning phase at a later time t_2 should be added to *i*'s BB at t_2. Axiom 1 would then ensure that *f* is a belief at t_2.

9. $[cand(f,i,t_1) \wedge later(t_1,t_2) \wedge successful(i,f,t_2)] \rightarrow should\text{–}be\text{–}inserted(i,f,t_2)$

Now, since the interpretation of 'questioning' may vary according to the intended application of the model, the interpretation of 'successful' should vary accordingly. If a particular agent altogether lacks a questioning mechanism, we may simply assign 'true' to 'successful', since there would be no tests which a candidate formula should pass, beyond being a candidate. Of course, questions may arise about how exhaustive a questioning or consistency testing process must be before we may assign 'success' to a formula. However, these are practical issues which must be decided in particular applications.

2.2 Formal Properties of the Model

We now consider some of the formal properties of NIM. First, note that the elements of NIM, with minor syntactic alterations, may be represented in any standard first-order predicate calculus, known to be sound and complete. We use 'PC' to denote any such logic.

Proposition 1. NIM is consistent.

The consistency of NIM may be checked by transforming each of its axioms into clause form, using a standard clause-form transformation algorithm. The NIM axioms are consistent just in case the resulting clauses are jointly consistent. One may easily verify that these clauses are simultaneously satisfiable in a universe of one individual. If the universe contains but one individual, all variables in the clauses may be replaced by a unique constant denoting this individual. It is then trivial to assign truth values to the resulting grounded literals so that each of the clauses is rendered true by the given interpretation.

Definition. Let **T** be a first-order theory formed by adding to PC an encoding of the axioms of NIM in PC.

Proposition 2. **T** is sound and complete.

Proof. The soundness of **T** follows from the fact that its rules of inference are exactly those of PC, and all axioms in **T** are consistent. If PC contains axioms, they are universally valid, and thus consistent with the consistent set NIM, which was added to PC to obtain **T**. The completeness of **T** follows immediately from the completeness of PC. The inference rules of PC ensure that all logical consequence of the NIM axioms encoded in PC are derivable.

3. DISCUSSION

We now return to a difficulty previously mentioned, namely, that NIM includes an explicit truth predicate. Since our model does not include a *definition* of the truth predicate, it is not vulnerable to the kind of internal inconsistencies which Tarski [1936] describes. Nevertheless, paradoxes involving self-reference can still arise on a *given interpretation* of the predicates in NIM. It may be objected, therefore, that the existence of potentially paradoxical sentences renders NIM defective. However, Kripke has clearly demonstrated [1975] that *any* language rich enough to permit the *ascription* of truth to sentences is vulnerable to paradox. He further demonstrates that this vulnerability in no way presupposes syntactic or semantic ill-formedness, but can arise for perfectly innocent looking sentences when unusual empirical conditions exist. Of course, one can always banish truth ascriptions in a special-purpose language, but if we are ever to have an adequate model of *human cognition*, we will need to represent propositions such as, "What Mary believes is true".

Since we are ultimately concerned with adequate cognitive models, we have included a truth predicate. Thus, paradoxical sentences may arise. However, both Kripke [1975] and Perlis [1985] have shown that the inclusion of such sentences need not render a first-order language inconsistent. Kripke proposes a complex theory of truth according to which paradoxical sentences are simply not assigned truth values. The underlying intuition of Kripke's theory is that any ascription of truth to a sentence involves a *presupposition* that the sentence is grounded. (Roughly, a grounded sentence is one whose truth evaluation "bottoms out", either immediately or ultimately, in the evaluation of sentences which do not ascribe truth or falsity to other sentences.) On the rare occasions when the presupposition of groundedness is unsatisfied no truth value is assigned. Kripke goes on to argue (following Kleene [1952]) that the existence of sentences without truth values does not require that we abandon any of the usual laws of logic. For example, since "$p \vee \neg p$" still holds for all cases when p *has* a truth value, we may retain the law. When p lacks a truth value, the disjunction is *not false*; rather it also lacks a truth value.

4. CONCLUSION

A generalized first-order model of belief and knowledge has been presented which simultaneously avoids logical omniscience and the excessively fine-grained "syntactic approach". The model, NIM, is intensionally based, and sanctions inferences involving nested epistemic attitudes, with different agents and different times. NIM makes explicit many (if not all) of the empirical assumptions an agent must make before concluding that another agent believes (knows) something *at a given time*. Because the model is axiomatized in FOL, it may be used by any standard, first-order theorem prover. Moreover, NIM provides agents with a conceptual map, interrelating the concepts of knowledge and belief and a number of cognate concepts, such as 'infers', 'retracts', and 'questions'. Because the model builds upon the concept of an *intension*, whose degree of granularity is left open, the range of formulas which are permitted to *express the same belief* is left to the user of NIM. For example, *one could* define "same intension" so that all logically equivalent formulas would have the same intension, and thus (by axiom 2) would express the same belief. However, if one wishes to retain an intuitive, human-like concept of belief, one

should individuate intensions so that (roughly) only those formulas which we intuitively regard as *synonymous* are permitted to have the same intension, and thus express the same belief. We have suggested several semantic theories which yield (approximately) the required criterion of synonymy.

REFERENCES

Cresswell, M.J., (1985) *Structured Meanings: The Semantics of Propositional Attitudes*, MIT Press, Cambridge, MA.

de Kleer, J. (1986) "An Assumption-based TMS", *Artificial Intelligence*, Vol. 28, pp. 127-162.

Doyle, J. (1979) "A Truth Maintenance System", *Artificial Intelligence*, Vol. 12, pp. 231-272.

Fagin, R., and Halpern, J., (1985) "Belief, Awareness, and Limited Reasoning", Proceedings of IJCAI, Los Angelos, pp. 491-501.

Frege, G. (1952) "On Sense and Reference", in *Translations from the Philosophical Writings of Gottlob Frege*, Oxford.

Gawron, J.M. (1986) "Types, Contents, and Semantic Objects", *Linguistics and Philosophy*, Vol. 9, No. 4, pp. 427-476.

Haas, A., (1985) "Possible Events, Actual Events, and Robots". *Computational Intelligence*, Vol. 1, No. 2, pp. 59-70.

Hadley, R.F. (1988) "Logical Omniscience, Semantics and Models of Belief", *Computational Intelligence*, Vol. 4, No. 1.

Hadley, R.F., (1988b) "A Default-Oriented Theory of Procedural Semantics", *Cognitive Science*, to appear.

Hadley, R.F., (1986) "Fagin and Halpern on Logical Omniscience: a Critique with an Alternative", Proceedings Sixth Canadian Conference on Artificial Intelligence, Montreal.

Hobbs, J.R., and Rosenschein, S.J., (1978) "Making Computational Sense of Montague's Intensional Logic", *Artificial Intelligence*, Vol. 9, pp. 287-306.

Kleene, S.C., (1952) *Introduction to Meta-Mathematics*, Wolters-Noordhoff Publishing and North-Holland Publishing, New York.

Konolige, K., (1985) "A Computational Theory of Belief Introspection", Proceedings of IJCAI, Los Angelos, pp. 503-508.

Lakemeyer, G., (1987) "Tractable Meta-Reasoning in Propositional Logics of Belief", Proceedings of IJCAI, Milan, pp. 401-408.

Levesque, H.J., (1984) "A Logic of Implicit and Explicit Belief", FLAIR Tech. Report #32, Fairchild, Palo Alto, Calif.

Lewis, D. (1976) "General Semantics", in *Montague Grammar*, (ed.) Partee, B., Academic Press, New York.

Maida, A.S. (1986) "Introspection and Reasoning about the Beliefs of Other Agents", *Proceedings of the Cognitive Science Society*, pp. 187-195.

Montague, R., (1970) "Universal Grammar", *Theoria* 36.

Perlis, D., (1985) "Languages with Self-Reference I: Foundations", *Artificial Intelligence*, Vol. 25, pp. 301-322.

Tarski, A. (1936) "Der Warheitsbegriff in den formalisierten Sprachen", *Studia Philos.*, Vol. 1, pp. 261-405.

Wilks, Y. & Ballim, A. (1987) "Multiple Agents and the Heuristic Ascriptions of Belief", Proceedings of IJCAI, Milan, pp. 118-124.

Subcognitive Probing: Hard Questions for the Turing Test

Robert M. French
Electrical Engineering and Computer Science Department
University of Michigan

INTRODUCTION

Alan Turing in his original article about an intelligence-game definition of intelligence seems to be making two separate claims. The first, call it the philosophical claim, is that if a machine could pass the Turing Test, it would necessarily be intelligent. I completely agree with this first claim. His second point, which I call the pragmatic claim, is that in the not-too-distant future it would, in fact, be possible to actually build such a machine. Turing clearly felt that it was important to establish both claims. He realized, in particular, that if one could rigorously show that no machine could ever pass his test, his philosophical point, while still true, would lose a great deal of its significance. He thus devoted considerable effort to establishing not only the philosophical claim but also the pragmatic claim.

Ever since his article appeared philosophers have concentrated almost exclusively on attacking or defending the philosophical claim. There are those who believe that passing the Turing Test constitutes a sufficient condition for intelligence and those who do not. As I said above, I wholeheartedly endorse the point of view that anything that could pass the Turing Test would be, without question, intelligent[2]. However, in this paper I will take issue with the pragmatic claim and argue that there is a flip side to Turing's extremely elegant test, namely, that its very capacity to probe the deepest, most essential areas of human cognition makes it, in a pragmatic sense, far too strong. The Turing Test could be passed only by things that have experienced the world as we have experienced it; the Test therefore provides a guarantee not of intelligence but of culturally-oriented *human* intelligence.

I establish this consequence of the Turing Test by proposing a class of questions, which I call *explicitly subcognitive questions*, that are intentionally designed to reveal low-level cognitive structure. Critics might object that there is something unfair about this type of question. This leads to the central idea of this paper which is that, in fact, there is *no way to distinguish questions that are subcognitive from those that are not.* To support this claim, I present another class of questions -- *implicitly* subcognitive questions -- that give every appearance of being at the cognitive level but that, in reality, are every bit as dependent on unconscious mechanisms as the initial class of explicitly subcognitive questions. In fact, close examination of some of the questions posed in Turing's original article reveals that they, too, are implicitly subcognitive. In like manner, any sufficiently broad set of questions making up a Turing Test would necessarily contain implicitly subcognitive questions. I show that it is impossible to tease apart implicitly subcognitive questions from explicitly subcognitive ones. And from this it follows that the cognitive and subcognitive levels are inextricably intertwined.

It is this essential inseparablility of the cognitive and subcognitive levels that, in a sense, undermines the Turing Test, making it too strong for its own good. As a result, it turns out to be a test for *human* intelligence, not intelligence in general. This fact, while admittedly interesting, is not particularly useful if our goal is to gain insight into intelligence in general. This seems to bring us back to the problem that Turing had hoped to sidestep by his imitation-game definition of intelligence, namely, the problem of specifying a set of necessary and sufficient conditions for intelligence. Unfortunately, there seems to be no easy way out; capturing the essence of general intelligence must be based on categorization and segmentation abilities, the ability to learn new concepts, the ability to adapt old ones to a new environment, and so on. Precisely what should be

on this list and what the definition of each of its components is, as it was in 1950, still unknown.

ON NORDIC SEAGULLS

Consider the following thought experiment: Suppose that the only flying animals known to the inhabitants of a large Nordic island are seagulls. Everyone on the island acknowledges, of course, that seagulls can fly. One day the two resident philosophers on the island are overheard trying to pin down what "flying" is really all about.

Says the first philosopher, "The essence of flying is to move through the air."

"But you would hardly call this flying, would you?" replies the second, tossing a pebble from the beach out into the ocean.

"Well then, perhaps it means to remain aloft for a certain amount of time."

"But clouds and smoke and children's balloons remain aloft for a very long time. And I can certainly keep a kite in the air as long as I want on a windy day. It seems to me that there must be more to flying than merely staying aloft."

"Maybe it involves having wings and feathers."

"Penguins have both, and we all know how well they fly . . ."

And so on. Finally, they decide to settle the question by, in effect, avoiding it. They do this by first agreeing that the only example of objects that they are absolutely certain can fly are the seagulls that populate their island. They do, however, agree that flight has something to do with being airborne and that physical features such as feathers, beaks, and hollow bones probably are superficial aspects of flight. On the basis of these assumptions and their knowledge of Alan Turing's famous article about a test for intelligence, they hit upon the Seagull Test for flight. The Seagull Test is meant to be a very rigorous sufficient condition for flight. Henceforth, if someone says, "I have invented a machine that can fly," instead of attempting to apply any set of flight-defining criteria to the inventor's machine, they will put it to the Seagull Test. The *only* things that they will certify with absolute confidence as being able to fly are those that can pass the Seagull Test. On the other hand, they agree that if something fails the Test, they will not pass judgment; maybe it can fly, maybe it can't.

The Seagull Test works much like the Turing Test: Our philosophers have two three-dimensional radar screens, one of which tracks a real seagull; the other will track the putative flying machine. They may run any imaginable experiment on the two objects in an attempt to determine which is the seagull and which is the machine, but they may watch them only on their radar screens. The machine will be said to have passed the Seagull Test for flight if both philosophers are indefinitely unable to distinguish the seagull from the machine.

An objection might be raised that some of their tests might have nothing to do with flight. They would reply: "So what? We are looking for a sufficient condition for flight, not a *minimal* sufficient condition. Furthermore, we understand that ours is a very hard test to pass, but rest assured, inventors of flying machines, failing the Test proves nothing. We will not claim that your machine *cannot* fly if it fails the Seagull Test; it may very well. However, we, as philosophers, want to be absolutely certain we have a true case of flight, and the only way we can be sure of this is if your machine passes the Seagull Test."

Now, of course, the Seagull Test will rightly take bullets, soap bubbles, and snowballs out of the running. This is certainly as it should be. But helicopters and jet airplanes -- which *do* fly -- would also never pass it. Nor, for that matter, would bats or beetles, albatrosses or hummingbirds. In fact, under close scrutiny, probably only seagulls would pass the Seagull Test, and maybe only seagulls from the philosophers' Nordic island, at that. What we have is thus not a test for flight at all, but rather a test for flight as practiced by a Nordic seagull.

For the Turing Test, the implications of this metaphor are clear: the Turing Test admits of no degrees in its determination of intelligence, in spite of the fact that the intuitive human notion of

intelligence clearly does. Spiders, for example, have little intelligence, sparrows have more but not as much as dogs, monkeys have still more but not as much as eight-year-old humans, who in turn have less than adults. If we agree that the underlying neural mechanisms (e.g., Hebbian learning) are essentially the same across species, then we ought to treat intelligence as a continuum and not just as "something that only humans have". It is especially important in the study of artificial intelligence that researchers not treat intelligence as an all-or-nothing phenomenon.

SUBCOGNITIVE QUESTIONS

Before beginning the discussion of subcognitive questions, I wish to make a few assumptions that I feel certain Turing would have accepted. First, I will allow the interrogator to have an assistant. I also want to make explicit an assumption that, in Turing's article, is tacit, namely that the human candidate and the interrogator (and, in this case, her assistant) are from the same culture and that the computer will be attempting to pass as an individual from that culture. Thus, if ever the computer replies, "I don't speak English" or something of the sort, the interrogator will immediately deduce, rightly, that the other candidate is the human being. Finally, while I believe that it is *theoretically* possible to build a machine capable of experiencing the world in a manner indistinguishable from a human being, I will assume that no computer is now, or will in the foreseeable future be, in a position to do so.

I will designate as *subcognitive* any question capable of providing a window on low-level (i.e., unconscious) cognitive structure. By "low-level cognitive structure", I am referring to the subconscious associative network in human minds that consists of highly overlapping activatable representations of experience. This is the level currently being explored by new approaches to cognitive modelling.[3]

The first type of subcognitive questions I will consider consists of those that I call *explicitly* subcognitive. I have chosen this name because these questions are explicitly designed to reveal low-level cognitive structure (and I think everyone would agree that they undeniably do so). I will respond to the anticipated objection that these explicitly subcognitive questions are unfair by following up with another set of questions that seem, at first glance, to be at a higher cognitive level than the first set. These questions will turn out, under closer examination, to be subcognitive as well. I will conclude with a final set of questions that seem for all the world to be innocent high-level cognitive questions but that will be just as hard as the others were for the computer to answer in the way a human would.

ASSOCIATIVE PRIMING

This first set of questions is based on current research on associative priming, often called semantic facilitation. The idea is the following: Humans, over the course of their lives, develop certain associations of varying strength among concepts. By means of the so-called lexical decision task it has been established[4] that it requires less time to decide that a given item is a word when that item is preceded by an associated word. If, for example, the item "butter" is preceded by the word "bread", it would take significantly less time to recognize that "butter" was a word than had an unassociated word like "dog" or a nonsense word preceded it.

The Turing Test interrogator makes use of this phenomenon as follows: She selects a set of words (and non-words), runs the lexical decision task on her assistant and records his recognition times. (This may, of course, be done prior to the start of the Turing Test. All that is required is that the interrogator come to the Test armed with the results of this initial test.) She then asks both candidates to perform the same experiment, and records their results. Once this has been done, she simply identifies as the human being the candidate whose results more closely resemble those produced by her assistant.

The machine would invariably fail this type of test because there is no *a priori* way of

determining associative strengths between *all* possible concepts. It would be necessary for the machine to know (or be able to determine) all of the associative strengths between all human concepts in order to pass this test without having experienced the world as the human candidate and the interrogator's assistant had. Unless the machine had had this experience, there would be a noticeably higher degree of priming similarity between the assistant and the human candidate than between the assistant and the machine.

Now, suppose a critic claims that these explicitly subcognitive questions are unfair because -- ostensibly, at least -- they have nothing to do with intelligence; they probe, the critic says, a cognitive level well below that necessary for intelligence and therefore they should be disallowed. Suppose, then, that we obligingly disallow such questions and propose in their stead a new set of questions that seem, at first glance, to be at a higher cognitive level.

THE RATING GAME

Neologisms will form the basis of the next set of questions I will propose. Our impressions involving made-up words provide particularly impressive examples of the "unbelievable number of forces and factors that interact in our unconscious processing of even . . . . words and names only a few letters long". [5]

I will now propose the following set of questions, all of which have a completely high-level cognitive appearance:

"On a scale of 0 (completely implausible) to 10 (completely plausible), please rate:

- 'Flugblogs' as a name Kellogg's would give to a new breakfast cereal.
- 'Flugblogs' as the name of a new computer company.
- 'Flugblogs' as the name of big, air-filled bags worn on the feet and used to walk on water.
- 'Flugly' as the name a child might give its favorite teddy bear.
- 'Flugly' as the surname of a bank accountant in a W.C. Fields movie.
- 'Flugly' as the surname of a glamorous female movie star.
- etc."

The interrogator will give, say, between fifty and one hundred questions of this sort to her assistant[6], who will answer them. Then, as before, she will give the same set of questions to the two candidates and compare their results to her assistant's answers. The candidate whose results most closely resemble the assistant's will, without doubt, be the human.

Let us examine a little more closely why a computer that had not acquired our full set of cultural associations would fail this test. Consider "Flugblogs" as the name of a breakfast cereal. It is unquestionably pretty awful. The initial syllable "flug" phonetically activates (unconsciously, of course) such things as "fug", "thug", "flub", "ugly", or "ugh!", each with its own aura of semantic connotations, while the second syllable, "blog", no doubt activates "blob", "bog", and other words, which in turn activate a halo of other semantic connotations. The sum total of this spreading activation determines how we react, at a conscious level, to the word. And while there will be no precise set of associated connotations for all individuals across a culture, on the whole there is enough overlap to provoke *similar* reactions to given words and phrases. In this case, the emergent result of these activations is undeniable: "Flugblogs" would be a lousy name for a cereal.

What about "Flugly" as a name a child might give its favorite teddy bear? Now *that* certainly sounds plausible. In fact, it's kind of cute. But, on the surface at least, "Flugblogs" and "Flugly" seem to have quite a bit in common; if nothing else, both words have a common first syllable. But "Flugly", unlike "Flugblogs", almost certainly activates "snugly" and "cuddly", which would bring to mind feelings of coziness, warmth, and friendship. It certainly also activates "ugly", which might normally provoke a rather negative feeling, but, in this case, there are competing positive associations of vulnerability and endearment activated by the notion of children and things that children like. To see this, we need look no further than the tale of the Ugly Duckling. In the end,

the positive associations seem to dominate the unpleasant sense of "ugly". The outcome of this subcognitive competition means that "Flugly" is perceived by us as being a cute, completely plausible name for a child's teddy bear. And yet, different patterns of activations rule out "Flugly" as a plausible name for a glamorous female movie star.

Imagine, for an instant, what it would take for a computer to pass this test. To begin with, there is no way it could look up words like "flugly" and "flugblogs"; they don't exist. To judge the appropriateness of any given word (or, in this case, nonsense words) in a particular context requires taking unconscious account of a vast number of culturally-acquired, competing associations triggered initially by phonetic resemblances. And, even though one might succeed in giving a program a certain number of these associations (for example, by asking subjects questions similar to the ones above and then programming the results into the machine), the space of neologisms is virtually infinite. The human candidate's reaction to such made-up words is an emergent result of myriad subcognitive pressures, and unless the machine had a set of associations similar to those of humans both in degree and in kind, its performance in the Rating Game would necessarily differ more from the interrogator's assistant's performance than would the human candidate's. Once again, a machine that had not experienced the world as we have would be unmasked by the Rating Game, even though the questions comprising it seemed, at least at the outset, so cognitively high-level in nature.

A VARIATION ON THE RATING GAME

If, for some reason, the critics were still unhappy with the Rating Game using made-up words, we could consider a variation on the game in which all of the questions would have the form:

"Rate **X**s as **Y**s" (0 = "could be no worse", 10 = "could be no better")

where **X** and **Y** are any two categories. Such questions give every appearance of being high-level cognitive questions: they are simple in the extreme and rely not on neologisms but on everyday words. For example, we might have, "Rate **radios** as **musical instruments**". Now, people do not usually think of radios as musical instruments, but they do indeed have some things in common with musical instruments: both make sounds; both are designed to be listened to; John Cage once wrote a piece in which radios were manipulated by performers; etc. There is therefore some overlap between the categories of **musical instruments** and **radios**. As a musical instrument, therefore, we might give a radio a rating of 3 or even 4 on a 10-point scale.

The answer to any particular rating question is necessarily based on how we view the two categories involved, each with its full panoply of associations, acquired through experience, with other categories. Other questions might be: "Rate **chocolate sundaes with nuts, whipped cream and cherries on top** as **antibiotics**", "Rate **grand pianos** as **wheelbarrows**", "Rate **purses** as **weapons**", "Rate **pens** as **weapons**", and so on. Just as before, it would be impossible to program into the machine all the various types and degrees of associations necessary to answer these questions like a human.

THE CENTRAL ISSUE

The central issue is that any reasonable set of questions in a Turing Test will necessarily contain subcognitive questions in some form or another. Ask enough of these questions and the computer will become distinguishable from the human because its associative network would necessarily be unlike ours. And thus the computer would fail the Turing Test.

Is it possible to modify the rules of the Turing Test in such a way that subcognitive questions are forbidden? I think not. The answers to subcognitive questions emerge from a lifetime of experience with the minutiae of existence, ranging from functionally adaptive world-knowledge to useless trivia. The sum total of this experience with its extraordinarily complex inter-relations is what defines human intelligence. And this is what Turing's imitation game tests for. What we

would really like is a test for intelligence *in general* -- but how could we achieve this through a Turing-like test? Surely, we do not want to limit ourselves to questions like, "What is the capital of France?" or "How many sides does a triangle have?". If we admit that intelligence in general must have *something* to do with categorization, analogy-making, and so on, we will, of course, want to ask questions that test these capacities. But the only questions that probe these capacities are, as I hope to have shown, subcognitive questions -- and we have seen where *those* questions lead!

CONCLUSION

In conclusion, the imitation game proposed by Alan Turing provides a very powerful means of probing human cognition. But its very strength is also, in a sense, a weakness. Turing invented the imitation game as a novel way of looking at the question "Can machines think?". However, the Turing Test is so powerful that it is really asking, "Can machines think exactly like human beings?". And this is less interesting than the first question. The Turing Test provides a sufficient condition for human intelligence but does not address the more important issue of intelligence in general.

I believe I have shown that only a computer that had acquired adult human intelligence by experiencing the world as we have could pass the Turing Test. In addition, I feel that any attempt to "fix" the Turing Test so that it could test for intelligence in general and not just human intelligence is doomed to failure because of the completely interwoven nature of human subcognition and cognition. To gain insight into intelligence, we will be forced to consider it in the more elusive terms of the ability to categorize, to generalize, to make analogies, to learn, and so on. It is with respect to these abilities that the computer will always be unmasked if it has not experienced the world as a human being has. In the final analysis, the Turing Test, as subtle and elegant as it is, still leaves us with the need to define general intelligence in terms of these abilities.

ACKNOWLEDGMENTS

I especially wish to thank Daniel Dennett for his invaluable comments on the ideas and emphasis of this paper. Douglas Hofstadter also had a major influence on the ideas of this paper; in particular, he suggested the distinction between implicitly and explicitly subcognitive questions, as well as some of the ideas on the Rating Game and its Variation. I would also like to thank Daniel Andler, Gray Clossman, Daniel Defays, Pierre Jacob, Melanie Mitchell, David Moser, and François Récanati for their suggestions. This research has been supported by a grant from the University of Michigan, a grant from Mitchell Kapor, Ellen Poss, and the Lotus Development Corporation, a grant from Apple Computer, Inc., and grant DCR 8410409 from the National Science Foundation.

ENDNOTES

[1]Turing, Alan M. (1950), Computing machinery and intelligence, Mind, Vol. 59, No. 236, pp. 433-460.

[2] For a particularly clear defense of this view see: Dennett, D. C. (1985). "Can Machines Think?" How We Know, ed. Michael Shafto. San Francisco, CA: Harper & Row.

[3]Three different approaches that all address subcognitive issues can be found in:

Feldman, J. and F. Ballard (1982). Connectionist models and their properties. Cognitive Science, 6(3), 205-254;

Hofstadter, D. R., M. Mitchell, and R. M. French (1987). Fluid concepts and creative analogies: A theory and its computer implementation. CSMIL Technical Report No. 10, University of Michigan;

Rumelhart, D. and J. McClelland (1986) (Eds.). Parallel Distributed Processing. Cambridge, MA: Bradford/MIT Press

[4]A particularly relevant, succinct discussion of associative priming can be found in: Anderson, J. R. (1983). The Architecture of Cognition. Cambridge, MA: Harvard University Press, Chap. 3, pp. 86-125. In this chapter Anderson makes reference to the classic work on facilitation by Meyer and Schvaneveldt (Meyer, D. E. and R. W. Schvaneveldt (1971). Facilitation in recognizing pairs of words: evidence of a dependence between retrieval operations. Journal of Experimental Psychology 90, 227-234).

[5] Hofstadter, D. R. (1984). On the seeming paradox of mechanizing creativity. In Metamagical Themas (pp. 526-546). New York, NY: Basic Books, Inc.

[6]Even though Turing did not impose a time constraint in his original formulation of the imitation game, he did claim that "...in fifty years' time [i.e., by the year 2000] it will be possible to programme computers . . . to make them play the imitation game so well that an average interrogator will not have more than 70 per cent. chance of making the right identification after five minutes of questioning" [p. 442]. In current discussions of the Turing Test, the duration of the questioning period is largely ignored. In my opinion, one reasonable extension of the Turing Test would include the length of the questioning period as one of its parameters. In keeping with the spirit of the original claim involving a five-minute questioning period, I have tried to keep the number of questions short although it was by no means necessary to have done so.

THE PRAGMATICS OF EXPERTISE IN MEDICINE

David A. Evans
Departments of Philosophy and Computer Science
Carnegie Mellon University

INTRODUCTION

A central goal of cognitive science research in problem solving has been to illuminate the nature of expertise (Chi & Glaser, in press; Johnson, et al., 1981). Some of this research has centered on the acquisition of specific skills that are prerequisites of expert performance (Feltovich, et al., 1984); some, on the differences in knowledge—in scope and organization—found among experts, intermediates, and novices (Chase & Simon, 1973; Chi, Feltovich & Glaser, 1981; Larkin, et al., 1980; Patel, 1984; Patel & Frederiksen, 1984).

The picture of expertise that emerges from these studies is of individuals who *know more* than non-experts; but whose knowledge differs not only in *quantity* but also *quality*. In particular, the knowledge of such experts is *dense* and *context-sensitive*: experts not only seem to have *more* domain-specific concepts, but more relations among concepts; and experts have a greater facility in identifying appropriate *contexts* in which to interpret the data in a problem (Evans, Gadd & Pople, in press).

In a domain like medicine, where problems may involve complex details and solutions may be grounded in incomplete models of pathophysiological processes, knowing more can lead to a tension between the need for both effective problem-solving behavior and thorough investigation of rival hypotheses. How such tensions are resolved reveals aspects of the pragmatics of medical expertise. This paper identifies two associated phenomena resulting from domain pragmatics—*selective ignorance* and *disciplinary ecological equilibrium*—that account for apparently sub-optimal behavior in medical expert problem solving.

ON PRAGMATICS IN MEDICINE

Pragmatics generally refers to the collection of effects in the processing of information that do not derive alone from either the *surface data* or the *semantics of the code* in the apprehended message. This characterization applies equally to linguistic and non-linguistic communications. Thus, pragmatics subsumes the inferences, interpretations, grounding of references, and other effects of context—whether the 'visible' co-occuring phenomena that accompany a communication or the 'invisible' mental models—that are mobilized in the pursuit of interpretations.

Pragmatics in medicine deserves special attention because successful problem solving depends on identifying the appropriate contexts in which to interpret the typically overwhelming number of possible features in any particular medical case. Such contexts are discovered by organizing surface observations (e.g., symptoms, signs, laboratory data) into patterns suggestive of diagnostic hypotheses; but the patterns themselves reflect components of disease models—idealizations of pathophysiological processes, often expressed as pseudo-causal relations among individual observations in a case. Since the available data in most

medical cases are too numerous, and the available correlations too complex, to support direct inductive reasoning, physicians must rely on *a priori* models that may be underdetermined by the immediately accessible information (Evans, Gadd & Pople, in press). One natural result is that physicians *hypothesize* candidate diagnoses; and one effect is that candidate diagnoses define contexts in which data may be *over-interpreted.*

ON EXPERTISE IN MEDICINE

Experts in medicine are individuals who have achieved a high level of competence, perhaps as measured by peer assessment—individuals who have completed all *normal* phases of training and have been practicing medical problem solving for some time. Typically, such experts are not expert across-the-board, in all phases of medical problem solving, but are 'specialists'—experts in solving domain-specific problem types, e.g., problems in internal medicine or pathology. But though specialized, medical expertise is not special-case or extraordinary expertise—is not cognitively anomalous—rather, is attainable by normally intelligent individuals. In particular, medical non-experts (novices, intermediates) are not cognitively immature individuals or laypeople. They are medical students, interns, and residents: individuals who have attained a high level of general competence and some degree of specialized training. The focus of research in medical problem solving is on how such non-experts acquire *domain-specific* knowledge and competence.

One measure of expertise is the ability to manage an amount of information that would overwhelm the novice. Indeed, that may be the single most striking and consistent characteristic of experts across subdomains of medicine. But there are aspects of medical expert behavior (and expert/novice differences) that indicate that such expert ability to manage data may not lead to commensurably improved reasoning abilities in problem-related tasks. For example, experts make accurate diagnoses even when they *explain* their diagnoses incorrectly (Patel & Groen, 1986). This suggests that different relations among the features in a case may be involved in actual problem solving than are revealed in the normative models of pathophysiological processes that form the community standard for medical explanations. As a related example, experts in one medical domain have difficulty solving problems in another medical domain, even when presented with the *basic science* information required to account for the pathophysiological processes involved in the problem (Patel, Arocha & Groen, 1987; Patel, Evans & Groen, in press). This suggests that the inferences that experts make are not based on 'first-principle' reasoning, but are guided by other, domain-specific, reasoning processes. As a final example, experts are typically more intolerant of uncertainty than novices, though they have the knowledge required to see a greater number of possibilites in interpreting data (Patel, Evans & Kaufman, in press). This suggests that the processing of information by medical experts is at least partly *categorial*, and that the categories effectively filter variation in data.

In short, expert performance is not perfect; and aspects of expert performance fall short of standards that experts, themselves, may acknowledge as desirable. (This is especially

the case in the emphasis on basic science as a basis for medical reasoning.) More problematic than simple failures in performance, though, is the fact that the observed behavior of experts—in their approach to problems, in their explanations—is consistent, coherent, and superficially rational. What can account for this?

EFFECTS OF PRAGMATICS IN MEDICAL PROBLEM SOLVING

If we take the above observations at face value, we must conclude that medical experts are *not* capable of flawless judgment, even in their areas of specialization; that, in a sense, they practice *false* science, or at least a different science than they imagine. In either case we should be concerned. If there is no method at all, there is no basis for the decisions physicians make. And if there is a method, but it is different from the one used to inform the discipline, there is no basis for predicting the effects of change—modifications of the normative models in the *façade* method—on actual practice.

On the surface, this has the form of a dilemma. I suggest, however, that it is the natural consequence of problem-solving behavior 'in the limit'; and that this situation may actually reflect an optimal adaptation to the needs of communication in complex domains. In particular, I would claim that the observed sub-optimal performance on some problem-related tasks can be attributed to the pragmatics of information management in medicine. One pragmatic effect is *local* (and 'low-level', in actual problem solving): the need to discriminate phenomena so that they can be analyzed as elements of complex categories. This leads to *selective ignorance*—a failure to see variation and to distinguish 'source' data from interpretations. Another pragmatic effect is *global* (and can have consequences for perfomance at all levels): the need to organize information in problem-solving situations is greater than the need to establish accurate relations among elements. This leads to an emphasis on principles of organization and on compliance with established standards of coherence (at the expense of principles of science) giving rise to *disciplinary ecological equilibrium*—where degredations in community standards pose a greater threat to performance than degredations in models. The following sections offer further reflections on these points.

Selective Ignorance

One source of error in expert performance derives directly from a source of efficiency: experts may fail to see certain relations in data because invoked, candidate interpretations effectively account for many of the relations in the data, obscuring possible alternative interpretations and inviting an under-interpretation of anomalous data. This is seen, for example, in studies of novices, intermediates, and experts in doctor/patient interviews (Patel, Evans & Kaufman, in press). The principal problem for novices is *seeing too much*—they cannot discriminate the relevant from irrelevant details; and they cannot organize all the information they gather in interviews (thought they may retain it). The principal problem for experts, on the other hand, is that they may *mis-manage* the flow of information by over-interpreting possible cues and, consequently, ignoring possibly important details.

In actual practice, the salvation in the first case is knowledge; in the second, discipline.

Through knowledge, the novice acquires prototypes that serve both as targets for the interpretation of details and also as predictors of additional relations or details that may be relevant. Prototypes become signposts on the way to the successful organization of the information in a case; and prototypes are *learnable*. Through discipline, the expert checks the tendency to premature closure—in particular, by following procedures that will typically require that details be examined from several points of view before a conclusion can be drawn. This forces the expert to re-examine the preferred hypothesis, though it may not force him or her to abandon it in the face of counterevidence.

It is interesting to note that experts we have studied do not recover well from defeated hypotheses. They frequently discard methodology at such points and generate inferences that are based on possible associations of data—not necessarily probable or plausible ones. Indeed, attempts to appeal to 'basic scientific' reasoning at such times is actually counterproductive (Patel, Evans & Groen, in press).

This may be partially explained by a related phenomenon, viz., that expert understanding involves what appears to be simultaneous derivation of a *situation model* along with the processing of *surface data*—and that the two are integrated in recall (Patel, 1988). This is distinct from the results with novices, where there appears to be a separation in memory of the facts and the derived situation.

Both phenomena—the over-interpretation of cues (and consequent under-interpretation of variation) and the integration of source data and interpretations—bear a striking similarity to what we see in *categorial perception*—where discrete phenomena are <u>not</u> noticed if they fall below a *threshold of perception*; but are homogenized and integrated into a *perceptual* or *processing category*, if they exceed the threshold. Typically, in such cases of perception, the source stimulus is lost and only the category and its prototypical predictions are recoverable.

In the face of noisy or complex data, categorial perception gives rise to efficient performance—but also selective ignorance: one has the ability to make clear judgments but one loses the ability to make (or see) fine distinctions. One important effect of selective ignorance is that all people who share the perceptual categories will agree on their classifications of phenomena. (This phenomenon assures agreements in phonological interpretations, for example, among native speakers of the same language.) But another effect is that when categorization proves incorrect, it may be difficult to see data objectively, without partial interpretations.

Disciplinary Ecological Equilibrium

A series of studies on the basis of medical-concepts have recently suggested that actual medical expertise may in some instances be based on *false* information (Coulson, Feltovich & Spiro, 1986; Feltovich, Spiro & Coulson, in press; Spiro, Feltovich & Coulson, 1987). Such observations clearly present a challenge to the science of medicine, but also to our understanding of effective problem solving. If the acknowledged basis for problem solving is flawed, what accounts for the persistence in use of inaccurate models in medcial textbooks,

in medical curricula, and in expert discourse? Is this one source of the sub-optimal behavior we observe in expert problem solving?

One analysis is that false models arise, in part, through simplifications that facilitate learning. The strong claim in this case would be that such practice leads to a *pernicious degradation* in the validity of a science. In the worst case, the cycle becomes: Learning *efficiency* leads to performance *deficiency*, which later leads to *new* learning *inefficiency*. More concretely, the simplifications that make the learning of complex concepts easier on first exposure actually become the bases of models that are flawed. These, in turn, give rise to performance anomalies; and inhibit the acquisition of new—and possibly 'correct'—models of biomedical phenomena.

An alternative interpretation is possible. Simplifications (even falsifications) are both adaptive and necessary in complex domains such as medicine. The claim would be that the need for coherence outweighs the need for accuracy; and coherence is assured only if the normative models are widely accepted, remembered, and used as the basis for the description, analysis, and discussion of problems. To the extent that detail is complex and renders models difficult to learn and use, actual accuracy can lead to degradations in agreement on coherence standards. Rather than promote performance deficiency, changes in practice (e.g., emphasizing scientific accuracy when we train physicians) may present a greater threat of reducing global expertise than hope of improving disciplinary performance.

The tension here is between the need for global coherence and problem solving efficiency (purchased via selective ignorance), on the one hand, and accuracy of knowledge (purchased via attention to greater detail—and perhaps intolerance of variation), on the other. The claim would be that in domains where perfomance grows asymptotic in the limit—and, more importantly, where expertise is *collective* in practice, not individual—there is a disciplinary ecological equilibrium—in which the need for great sensitivity to coherence and great conformity with paradigms (to promote agreement in problem solving behavior) is more important than accuracy of the accepted models. In such a case, the superfical biases we observe would actually be the symptoms of constructive adaptive behavior.

CONCLUSION

The source of the problems we observe in medical experts derives from limitations in human intelligence (and epistemological models) in capturing the ontology of biomedicine. The uniformity of performance across experts reflects the 'sociology' of the discipline—the adaptive behavior associated with shared and reinforced principles of communication and explanatory adequacy.

One lesson is that what experts do well may necessarily determine what experts do poorly. The medical expert purchases efficiency with selective ignorance; and achieves effective communication and coherence through a conspiracy of simplifications. These are two of the pragmatic features of expertise in medicine.

The reflections offered here require elaboration beyond the scope of this paper; but they

raise questions that have implications for our understanding of learning and skill mastery in other complex problem-solving domains—not just in *medicine*.

REFERENCES

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.

Chi, M.T.H. & Glaser, R. (Eds.) (In Press). *The nature of expertise.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Coulson, R.L., Feltovich, P.J. & Spiro, R.J. (1986). Foundations of a misunderstanding of the ultrastructural basis of myocardial failure: A reciprocating network of oversimplifications (Report No. 1). Springfield, IL: Conceptual Knowledge Research Project, Southern Illinois University School of Medicine.

Evans, D.A., Gadd, C.S. & Pople, H.E., Jr. (In Press). Managing coherence and context in medical problem-solving discourse. In D.A. Evans & V.L. Patel (Eds.), *Cognitive science in medicine.* Cambridge, MA: The MIT Press.

Feltovich, P.J., Johnson, P.E., Moller, J.H. & Swanson, D.B. (1984). LCS: The role and development of medical knowledge in diagnostic expertise. In W.J. Clancey & E.H. Shortliffe (Eds.), *Readings in medical artificial intelligence: The first decade.* Reading, MA: Addison Wesley.

Feltovich, P.J., Spiro, R.J. & Coulson, R.L. (In Press). The nature of conceptual understanding in biomedicine: The deep structure of complex ideas and the development of misconceptions. In D.A. Evans & V.L. Patel (Eds.), *Cognitive science in medicine.* Cambridge, MA: The MIT Press.

Johnson, P.E., Duran, A.S., Hassebrock, F., Moller, J., Prietula, M., Feltovich, P.J. & Swanson, D.B. (1981). Expertise and error in diagnostic reasoning. *Cognitive Science*, *5*, 235–283.

Larkin, J.H., McDermott, J., Simon, D.P. & Simon, H.A. (1980). Expert and novice performance in solving physics problems. *Science*, *208*, 1335–1342.

Patel, V.L. (1984). *Prior knowledge and comprehension of medical texts: Expert-novice differences.* Centre for Medical Education, McGill University, Montreal, Canada. CME Report # CME84-CS3.

Patel, V.L. (1988). *The nature of understanding and problem solving in medical expertise.* Centre for Medical Education, McGill University, Montreal, Canada. CME Report # CME88-CS4.

Patel, V.L., Arocha, J.F. & Groen, G.J. (1986) Strategy selection and degree of expertise in medical reasoning. *Proceedings of the Eight Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Patel, V.L. & Frederiksen, C.H. (1984). Cognitive processes in comprehension and knowledge acquisition by medical students and physicians. In H.G. Schmidt & M.C. de Volder (Eds), *Tutorials in problem-based learning.* Assen, Holland: Van Gorcum.

Patel, V.L. & Groen, G.J. (1986). Knowledge-based solution strategies in medical reasoning. *Cognitive Science*, *10*, 91–115.

Patel, V.L., Evans, D.A., & Groen, G.J. (In Press). Biomedical knowledge and clinical reasoning. In D.A. Evans & V.L. Patel (Eds.), *Cognitive science in medicine*. Cambridge, MA: The MIT Press.

Patel, V.L., Evans, D.A. & Kaufman, D. (In Press). A cognitive science framework for doctor/patient interaction. In D.A. Evans & V.L. Patel (Eds.), *Cognitive science in medicine*. Cambridge, MA: The MIT Press.

Spiro, R.J., Feltovich, P.J. & Coulson, R.L. (1987). *Seductive reductions: The hazards of oversimplification of complex concepts* (Report No. 4). Springfield, IL: Conceptual Knowledge Research Project, Southern Illinois University School of Medicine.

COGNITIVE FLEXIBILITY THEORY: ADVANCED KNOWLEDGE ACQUISITION IN ILL-STRUCTURED DOMAINS

Rand J. Spiro
University of Illinois at Urbana-Champaign

Richard L. Coulson, Paul J. Feltovich, and Daniel K. Anderson
Southern Illinois University School of Medicine

Advanced knowledge acquisition in a subject area is different in many important ways from introductory learning (and from expertise). In this paper we discuss some of the special characteristics of advanced learning of complex conceptual material. We note how these characteristics are often at odds with the goals and tactics of introductory instruction and with psychological biases in learning. We allude to our research in biomedical cognition that has revealed a substantial incidence of misconception attributable to various forms of oversimplification, and we outline the factors that contribute to suboptimal learning at the advanced stage. We then sketch a theoretical orientation for more successful advanced knowledge acquisition in ill-structured domains, Cognitive Flexibility Theory.

The Goals of Advanced Knowledge Acquisition

In our work we have been interested in "advanced knowledge acquisition"--learning beyond the introductory stage for a subject area, but before the achievement of practiced expertise that comes with massive experience. This often neglected intermediate stage is important because the aims and means of advanced knowledge acquisition are different from those of introductory learning. In introductory learning the goal is often mere exposure to content and the establishment of a general orientation to a field; objectives of assessment are likewise confined to the simple effects of exposure (e.g., recognition and recall). At some point in learning about a knowledge domain the goal must change; at some point students must "get it right." This is the stage of advanced knowledge acquisition [7, 10, 12]: the learner must *attain a deeper understanding of content material, reason with it, and apply it flexibly in diverse contexts*. Obstacles to advanced knowledge acquisition include conceptual complexity and the increasing ill-structuredness that comes into play with more advanced approaches to a subject area. By *ill-structuredness* we mean that many concepts (interacting contextually) are pertinent in the typical case of knowledge application, and that their patterns of combination are inconsistent across case applications of the same nominal type. (See [12] for a more detailed treatment of the nature and consequences of ill-structuredness.)

The methods of education in introductory and advanced learning seem, in many ways, to be at odds. For example, compartmentalizing knowledge, presenting clear instances (and not the many pertinent exceptions), and employing reproductive memory criteria are often in conflict with the realities of advanced learning--knowledge which is intertwined and dependent, has significant context-dependent variations, and requires the ability to respond flexibly to "messy" application situations. These discrepancies in aims and tactics (along with many others that we have observed) raise the possibility that introductory learning, even when it is "successful," lays foundations in knowledge and in an approach to learning that interfere with advanced acquisition. As we have seen repeatedly demonstrated, that possibility is an actuality [6, 7, 10, 12].

Deficiencies in Advanced Knowledge Acquisition

Medical school is an archetype of an advanced knowledge acquisition setting [7]. Medical students have already had introductory exposure to many of the subject areas of biological science that they go on to study in medical school, but they are certainly not yet expert. Furthermore, the goals of medical education are clearly those of advanced knowledge acquisition. Important aspects of conceptual complexity must now be mastered (superficial familiarity with key concepts is no longer sufficient); and the ability to apply knowledge

from formal instruction to real-world cases is certainly something that is expected of those studying to be physicians.

In our laboratory we have been studying medical students' learning, understanding and application of important but difficult biomedical science concepts. This effort has revealed widely held systematic misconceptions among students, despite their having been exposed to appropriate information [6, 7, 10, 12]. Stubborn misconception, notwithstanding usual classroom efforts at instruction, has been found for difficult concepts in other areas as well (e.g., physics: [3, 13]).

The biomedical misconceptions that we have identified are of various kinds [7, 10]. These include *contentive* errors, often involving overgeneralization; for example, areas of subject matter are seen as being more similar than they really are. Errors attributable to dysfunctional biases in *mental representation* are also observed; for example, dynamic processes are often represented more statically. *Prefigurative "world views"* that underlie learners' understanding processes also cause problems; for example, the presupposition that the world works in such a way that "parts add up to wholes" leads students to decompose complex processes into components that are treated (mistakenly) as independent. Furthermore, at all these levels misconceptions interact in *reciprocally supportive* ways, and combine to yield higher order misconceptions [6, 7]. Failures of understanding *compound* themselves, building up durable chains of larger scale misconception.

Reductive biases: The pervasive role of oversimplification in the development of misconceptions. A predominant share of the misconceptions (and networks of misconception) that we have identified reflect one or another kind of *oversimplification* of complex material--associated with learners' earlier experiences with introductory learning, and even influenced by many experiences with advanced learning. Misconceptions of advanced material result both from interference from earlier, simplified treatments of that material and from a prevailing mode of approaching the learning process in general that fosters simplificational strategies and leaves learners without an appropriate cognitive repertoire for the processing of complexity [7, 10, 12].

We have termed the general tendency to reduce important aspects of complexity the *reductive bias*. Several forms of the bias have been identified, selected examples of which follow (see [6, 7, 10] for *examples of biomedical misconceptions* corresponding to the types of reductive bias listed):

1. Oversimplification of complex and irregular structure. Superficial similarities among related phenomena are treated as unifying characteristics. Interacting components are treated as independent. Incomplete conceptual accounts are presented (or accepted by the learner) as being comprehensive. Instances that are referred to as belonging to the same generic category are treated in a uniform manner despite their being highly diverse. The irregular is treated as regular, the nonroutine as routine, the disorderly as orderly, the continuous as discrete, the dynamic as static, the multidimensional as unidimensional. (This first reductive bias is the most general one, encompassing many of the specific ones listed below.)

2. Overreliance on a single basis for mental representation. A single, encompassing representational logic is applied to complex concepts and phenomena that are inadequately covered by that logic. For example: Understanding of a new concept is reduced to the features of a (partially) analogous concept. New, highly divergent examples are understood by exclusive reference to a single prototype. A single schema or theory is profferred and preferred, despite the fact that its coverage is significantly incomplete. Complexly multifaceted content has its understanding narrowed to just those aspects covered by a single organizational scheme. And so on.

3. Overreliance on "top down" processing. Understanding and decision making in knowledge application situations (i.e., cases) rely too exclusively on generic abstractions (i.e., concepts, theories, etc.); detailed knowledge of *case* structure is not used enough (i.e., knowledge of "how cases go," as well as reasoning from specific case precedents).

4. Context-independent conceptual representation. The contexts in which a concept is relevant are treated as having overly uniform characteristics. This promotes the representation of conceptual knowledge in a manner too abstract for effective application (i.e., without sufficient regard for the specifics of application in context). Concepts are insufficiently tailored to their uses; concepts are not recognized as relevant when, in fact, they are; and concepts are mistakenly judged to be relevant in contexts where they are not.

5. Overreliance on precompiled knowledge structures. *Fixed* protocols or rigidly *prepackaged* schemas are presented to learners and used by them as recipes for what to do in new cases.

6. Rigid compartmentalization of knowledge components. Components of knowledge that are in fact interdependent are treated as being separable from each other. Learners develop mistaken beliefs in the independence of the components. Relatedly, where knowledge components do function independently, it may nevertheless be the case that conveying relationships between their conceptual structures would aid understanding; these connections are not drawn. When components are interrelated, there is a tendency to use just one linkage scheme, thereby underrepresenting the richness of interconnection in the system and promoting narrow, doctrinaire viewpoints (see the problem of single representations).

7. Passive transmission of knowledge. Knowledge is preemptively encoded under a scheme determined by external authority (e.g., a textbook) or a scheme which facilitates delivery and use. Knowledge is "handed" to the learner. The preemptive encoding is passively received by the learner, and useful benefits that result from personalized knowledge representations, derivable from active exploration and involvement in the subject area, do not develop. When active, participatory learning is encouraged, adequate support for the management of increased indeterminacy and cognitive load is not provided (e.g., mentor guidance, memory aids, etc.).

The next section will outline our theoretical approach to *remedying* the problems of advanced knowledge acquisition that are caused by these reductive biases.

Cognitive Flexibility Theory: Themes of Advanced Knowledge Acquisition

Where has our research on the *problems* of advanced knowledge acquisition led us? To an overall theoretical orientation that in many ways derives its fundamental themes from the specific nature of those learning problems, as the problems relate to the characteristics of ill-structured domains and the special goals of advanced knowledge acquisition (i.e., mastery of conceptual complexity and knowledge application/transfer).

In this section we provide a brief discussion of our most fundamental, theoretically motivated remedies for the problems of advanced knowledge acquisition. The following themes constitute different facets of what we call *cognitive flexibility* [12]. The themes are, in a sense, *conditions* for developing mastery of complexity and knowledge transferability. Each of the headlined theoretical commitments has received some form of implementation, either in our experiments or in our theory-based computer hypertext systems (including one prototype that implements the theory's principles of advanced knowledge acquisition in cardiovascular medicine, the *Cardioworld Explorer*). Given the extreme limitations of space, the themes are discussed schematically and in the abstract; detailed development of theoretical rationales, examples of our concrete instantiations of the themes (in the biomedical domain and others that we have studied), and patterns of empirical support for our claims can be found in our cited papers.

1. Avoidance of Oversimplification and Overregularization. Because of the strong bias towards oversimplification that we have observed, it is clear that advanced knowledge acquisition must place a high premium on making salient those ways that knowledge is not as simple and orderly as it might first seem in introductory treatments. Where the problem is so often a presumption of simplicity and regularity, the remedy is to take special measures to *demonstrate complexities and irregularities*. It is important to lay bare the limitations of initial, first pass understandings, to highlight exceptions, to show how the superficially similar is

dissimilar and how superficial unities are broken. Where conceptual error frequently occurs from atomistic decomposition of complexly interacting information, followed by misguided attempts at "additive" reassembly of the decomposed elements, the remedy is to take pains to *highlight component interactions*, to clearly demonstrate the intricate patterns of *conceptual combination*.

This is a very general theme, encompassing many of the others that follow in this list. Cognitive flexibility involves the *selective* use of knowledge to *adaptively fit* the needs of understanding and decision making in a particular situation; the potential for maximally adaptive *knowledge assembly* depends on having available as full a representation of complexity to draw upon as possible.

2. Multiple Representations. Single representations (e.g., a single schema, organizational logic, line of argument, prototype, analogy, etc.) will miss important facets of complex concepts. Cognitive flexibility is dependent upon having a diversified repertoire of ways of thinking about a conceptual topic. Knowledge that will have to be used in many ways has to be learned, represented, and tried out (in application) in many ways.

The use of multiple representations is important at different levels. For example, we have found multiple analogies to be very useful in understanding complex individual concepts ([10]; see the example below of force production by muscle fibers; see also [5, 14]). However, the importance of multiple representations may be even more important for larger units of analysis. For example, we have found that students' understandings of the entire domain of biomedical knowledge is adversely affected by the tendency to use just one way of modeling the various phenomena they encounter, one that comes from the *metaphor of the machine*. This one "lens" leads them to take for granted certain issues related to the nature of explanations, the structure of mental models of functional systems, and so on. These students develop understandings that do not capture important aspects of the biomedical domain (e.g., inherently organic processes). Their understandings would be more complete if they were to augment the selective view that results from their mechanistic bias with other understandings that selectively emerge from the unique aspects of other cognitive "lenses," for example, from *organicist* metaphors [7].

The need for multiple representations applies not only to complex concepts, but to cases as well. In an ill-structured domain, cases (examples, occurrences, events--occasions of use of conceptual knowledge) tend to be complex and highly variable, one to the next. The complexity of cases requires that they be represented from multiple theoretical/conceptual perspectives--if cases are treated narrowly by characterizing them using a too limited subset of their relevant perspectives, the ability to process future cases will be limited. First, there will be an assumption that cases are simpler than they in fact are, and attempts to deal with new cases will prematurely conclude after they are only *partially* analyzed. Second, there will be insufficient preparedness to deal with the specific patterns of interaction of theoretical/conceptual perspectives within cases. Third, to the extent that performance in future cases will require reasoning from sets of precedent cases (which is always a greater need in ill-structured domains), the likelihood of having case representations available in prior knowledge which are maximally apt in their relation to some new case is lessened to the extent that cases are narrowly represented in memory. This is especially so when there is substantial across-case dissimilarity; the relative novelty of a new case in an ill-structured domain will require more elaborate efforts to find appropriate precedents--the wider the variety that is available, the better the chances of finding a fit.

An Example of Multiple Representations: Integrated Multiple Analogies for Complex Concepts. As we have said, our studies of medical students have indicated that one of the most serious contributors to the problems of advanced knowledge acquisition is the use of a single knowledge representation. Complex concepts can rarely be adequately represented using a single schema, theoretical perspective, line of exposition, and so on. Nevertheless, in practice, complex concepts frequently are represented in some single fashion, with substantial consequences.

Our remedy has been to approach learning in all of the domains that we have studied with the goal of promoting *multiple representations* (e.g., multiple precedent cases for a new case; multiple organizational schemes for representing the same content material in our computer hypertexts; etc.). Here we will briefly

consider just the case of analogy. We have discovered a large number of misconceptions that result from the overextended application of analogies [10]. To combat the negative effects of a powerful and seductive single analogy, we employ *sets of integrated multiple analogies*. Whenever a source concept in an analogy is missing important aspects of a target concept, or the source concept is in some way misleading about the target concept, we introduce *another* analogy to counteract those specific negative effects of the earlier analogy.

So, where we find that misconceptions about the nature of force production by muscle fibers often develop because of a common analogy to the operation of rowing crews (sarcomere "arms" and oars both generate force by a kind of "pulling"), other analogies are introduced to mitigate the negative effects of the limited rowing crew analogy [10]. An analogy to turnbuckles corrects misleading notions about the nature of relative movement and the gross structures within the muscle. An analogy to "finger handcuffs" covers important information missing in the rowing crew analogy about limits of fiber length (the elastin covering on muscle fiber bundles constricts at long lengths, stopping extension in a manner similar to the cross-hatched finger cuffs when you try to pull a finger out of each end). And so on. A composite imaging technique that helps the user to integrate the multiple analogies, so that the correct aspects of each analogy can be selectively instantiated in relevant contexts of use of the target concept, has also been developed. The procedure facilitates the learning of a concept (through the pedagogical benefits of analogy), while maintaining the integrity of the concept's complexities (by using multiple analogies to cover the concept's multifacetedness and to vitiate the force of incorrect aspects of any single analogy). (Also see [2].)

Theory-based hypertext systems to implement the themes of advanced knowledge acquisition in ill-structured domains: The importance of revisiting and rearranging in the development of multiple representations. Much of the work on computer hypertext systems has been driven by the power of the technology, rather than by a coherent view of the cognitive psychology of nonlinear and multidimensional learning and instruction. In contrast, our hypertext approaches have a basis in cognitive theory--they derive from the themes of Cognitive Flexibility Theory. And their realm of operation is specified; they are especially targeted at advanced knowledge acquisition in ill-structured domains. (There is no point in imposing the extra cognitive load of nonlinearity and multidimensionality if the domain being studied is simple and well-structured, or if the goals of learning are the more easily attainable ones of introductory treatments.) We will briefly characterize our approach to implementing Cognitive Flexibility Theory in computer hypertext systems.

Our hypertext systems build multiple representations in a manner that can be understood using a metaphor of landscape exploration. Deep understanding of a complex landscape will not be obtained from a single traversal. Similarly for a *conceptual* landscape. Rather, *the landscape must be criss-crossed in many directions* to master its complexity and to avoid having the fullness of the domain attenuated [12, 15]. The same sites in a landscape (the same cases or concepts in a knowledge domain) should be *revisited* from different directions, thought about from different perspectives, and so on. There is a limit to how much understanding of a complex entity can be achieved in a single treatment, in a single context, for a single purpose. By *repeating* the presentation of the same complex case or concept information in *new contexts*, additional aspects of the multifacetedness of these "landscape sites" are brought out, enabling the kind of rich representations necessary in a complex and ill-structured domain. Thus, cognitive flexibility is fostered by a flexible approach to learning and instruction. The same content material is covered in different ways, at different times, in order to demonstrate the potential flexibility of use inherent in that content ([11, 12]).

3. Centrality of Cases. The more ill-structured the domain, the poorer the guidance for knowledge *application* that "top-down" structures will generally provide. That is, the way abstract concepts (theories, general principles, etc.) should be used to facilitate understanding and to dictate action in naturally occurring cases becomes increasingly indeterminate in ill-structured domains. The application of knowledge to cases in an ill-structured domain (i.e., a domain in which cases are individually multidimensional, and irregularly related one to the next) cannot be prescribed in advance by general principles. This is because, in ill-structured domains, there is great variability from case to case regarding which conceptual elements will be relevant and in what pattern of combination. In an ill-structured domain, general principles will not capture

enough of the structured dynamics of cases; increased flexibility in responding to highly diverse new cases comes increasingly from reliance on reasoning from precedent cases.

Thus, examples/cases cannot be assigned the ancillary status of merely illustrating abstract principles (and then being discardable); the cases are key--examples are necessary, and not just nice [7, 11, 12].

4. Conceptual Knowledge as Knowledge-in-Use. Not only is it more difficult to count on top down prescriptions for performance in new cases in an ill-structured domain (i.e., abstract concepts/theories inadequately determine responses to new cases), but there is also considerable indeterminateness in defining conditions for *accessing* conceptual structures in the first place, to engage the guidance the conceptual structures *do* offer. It is not that abstract knowledge has no role in ill-structured domains, but that its role is highly intertwined with that of case-centered reasoning. Put another way, in an ill-structured domain there will be greatly increased variability across cases in the way the same concept is used or applied. Thus it is harder to get from features of cases to the concepts that might need to be applied to those cases. And it is harder to apply a concept, once accessed, if it has many different kinds of uses across cases--concepts must be *tailored* to their application contexts. The Wittgensteinian dictum that meaning is determined by use clearly applies in ill-structured domains. If a concept's meaning in use cannot be determined universally across cases (as in an ill-structured domain), then one must pay much more attention to the details of how the concept is used--*knowledge in practice*, rather than in the abstract [11, 12, 15].

In medical training, this issue of *variability and combination in concept instantiation* has an obvious implication for the traditional difficulty of integrating the biomedical basic science parts of the curriculum with the clinical parts. Physicians' practice would be improved if in problematic situations they could apply the interacting basic biomedical science concepts that underlie the clinical situation that is posing the problem. However, it is very difficult for medical students to learn how to get to the basic science concepts from clinical presenting features, partly because of the great variability across clinical cases in the way those concepts get instantiated. A key feature of our *Cardioworld Explorer* hypertext is that it permits the learner to selectively examine the full range of uses of any selected basic science concept (or any selected combination of concepts) across cases with differing clinical features, teaching the patterns of concept application and thus facilitating *access* to conceptual information in clinical contexts (as well as fostering an understanding of the different ways that a given concept has to be tailored to be clinically relevant).

Again, in an ill-structured domain the meaning of a concept is intimately connected to its patterns of use. When the uses (instances, cases) of the same concept have a complex and irregular distribution (i.e., the domain is ill-structured), adequate prepackaged prescriptions for proper activation of the concept cannot be provided (i.e., *concept instantiation is non-routine*). Instead, greater weight (than in a well-structured domain) must be given to activating concepts in a new case by examination of family resemblances across the features of *past cases* that have been called (labeled as instances of) that concept.

5. Schema Assembly (from Rigidity to Flexibility). In an ill-structured domain, emphasis must be shifted from *retrieval* of intact, rigid, precompiled knowledge structures, to *assembly* of knowledge from different conceptual and precedent case sources to adaptively fit the situation at hand [9, 12]. This follows, again, from characteristics of ill-structured domains. Since ill-structuredness implies kinds of complexity and irregularity that militate against the use of knowledge structures that assume routinizability across cases, the role of intact schema retrieval must be diminished--greater across-case differences cause a necessary decline in the ability of any large, single precompilation to fit a wide variety of cases. In complex and ill-structured domains, one cannot have a prepackaged schema for everything! As ill-structuredness increases, the use of *rigid* knowledge structures (i.e., the same precompiled knowledge structure used for many cases) must be replaced by *flexible*, recombinable knowledge structures. For any particular case, many *small* precompiled knowledge structures will need to be used. And there will be relatively little repetition of patterns across case-specific assemblies of these smaller pieces of precompiled knowledge. Accordingly, in knowledge acquisition for cognitive flexibility, the "storage of fixed knowledge is devalued in favor of the *mobilization of potential knowledge*" ([12], p. 181). (See also [8].)

6. Noncompartmentalization of Concepts and Cases (Multiple Interconnectedness). Because of the complex and irregular way that abstract conceptual features weave through cases/examples in ill-structured domains, knowledge cannot be neatly compartmentalized. In order to enable the situation-dependent, adaptive schema assembly from disparate knowledge sources that characterizes cognitive flexibility, those multiple sources must be highly interconnected. Concepts cannot be treated as separate "chapters." Retroactive assembly of independently taught, and noninterrelated, constituent conceptual aspects too often fails. Also, although cases have to be focused on separately, so that the complexity of case structure is conveyed, they should not be taught in *just* that way--connections across cases must also be established. Rather than relegating concepts or cases to separate compartments, chapters, and so on, our systems strive for *multiple interconnectedness (of cases and concepts) along multiple conceptual and clinical dimensions.*

Our approach to fostering multiple interconnectedness of knowledge representations in our hypertexts is to code case segments with a multidimensional vector indicating the relevance of a variety of thematic/conceptual dimensions to that case segment [11]. (Positive values in the vector also point to commentary, providing expert guidance about the nature of the conceptual dimension's instantiation in that particular case segment; this helps with the problem of teaching conceptual knowledge-in-use discussed earlier). Then, as the hypertext program guides the learner in criss-crossing the domain's "landscape," by exploring patterns of overlap in the vectors for different case segments, knowledge representations are built up in which parts of cases are connected with many parts of other cases, along many conceptual/theoretical dimensions of case-segment similarity. In that way, many alternative paths are established to get from one part of the overall knowledge base to any other part of the knowledge base that aspects of some future case may signal as relevant. Thus, the potential for flexible, situation-adaptive schema assembly is fostered (along with such other virtues as the establishment of multiple routes for memory access to any node in the system).

So, for example, in the *Cardioworld Explorer* segments of clinical cases are encoded with a vector of clinical and basic biomedical science themes that are relevant to each segment. The system can then establish connections between a segment of one case and segments of many other cases, along the various (conceptual and clinical) thematic dimensions represented in the vector. In case-based instruction, it is often true that there are important, instructive relationships between an aspect of one case and aspects of others. Such relationships are rarely brought out. Our hypertext systems capture these many lessons that are missed in strict case-by-case (or problem-by-problem) instruction. In an ill-structured domain, facilitating retrieval of multiple (partial) precedents is important, because understanding what to do in a given case context will usually require reference to more than any single prototype--the case in question will be "kind of like this earlier one, kind of like that one," and so on. Also, understanding of the case in question will require that various concepts be brought to bear and integrated; this, too, is facilitated by the multiple conceptual coding scheme employed in our systems.

There are several other benefits of the multiple-conceptual coding of multiple case segments. A power/efficiency advantage is that it allows the hypertexts to automatically generate large numbers of lessons (many "landscape criss-crossings"). If, for example, each of 20 cases is divided into an average of 10 case segments, each with a value on 15 relevant thematic dimensions, there is a many-fold increase in the number of possible automatizable instructional comparisons and contrasts that results from having 200 case segments (instead of 20 full cases) intertwined by relationships in the 15-slot vector.

Also, the use of case segments prevents the subsumption to a "common denominator" that occurs when larger structural units are used: an interesting local element of a case will tend to get lost if it has features that are not present in other parts of the case (when the monolithic case is the structural unit). Using small case segments (minicases) helps to retain the *plurality* of situations.

There is another virtue of the division into case segments and the multidimensional coding of the segments that relates to keeping case understanding from being overly simplified. In an ill-structured knowledge domain, by definition, there is sufficient variability across cases (due in part to the interaction of the many

factors that make up complex cases) that the set of cases that might be nominally grouped together under some schema or classification will be greatly variable in their particulars. A case, instead of being represented as one kind of thing, conveying one kind of "lesson," is instead clearly shown to the learner to to be *many things*. Cases of the same nominal type have different segments or scenes that are demonstrated not to be the same, and each of the segments is shown to have multiple significances. Therefore, the common temptation to nest cases uniquely under a single superordinate conceptual category will be resisted, making it less likely that the complex relationships among cases in a domain will be artificially regularized. In an ill-structured domain, cases are related to many different concepts of the domain, and it promotes dysfunctional simplification to hierarchically nest or "slot" cases under single conceptual categories (e.g., "The following cases are examples of X [only]"). When there is considerable across-case variability, as there will be in an ill-structured domain, cognitive flexibility requires that case information be coded conceptually for the many different kinds of use that new situations may require.

The thematic coding scheme and the landscape criss-crossing system of instruction result in a weblike multiple interconnectedness on multiple dimensions that is not subject to the limitations of instruction that is characterized by a single organizational slant. Instead of a single text with a single organizational scheme and a single sequencing of comparisons and contrasts, our hypertexts allow the same information to be automatically reconfigured according to a huge number of possible organizational schemes, determined by using subsets of the multiple thematic coding space--our hypertexts enable the *virtually limitless automatic generation of new text configurations*. Because of the richness of ill-structured domains such as biomedical science, each of these text configurations teaches some case- (experience-) grounded lessons that would not have been taught (or easily seen if taught) from another text's organizational perspective. Such additional experiences and perspectives are always helpful in a complex domain--a physician never learns all that it would be helpful to learn (which is why additional experience is always valued in a physician). Hypertext systems like the *Cardioworld Explorer* systematically *consolidate the process of acquiring experience*.

Yet another virtue of the multiple interconnectedness along multiple dimensions of the representations that our systems build has to do with the problem of reciprocal misconception compounding that we have observed in our studies of medical students and physicians [6, 7]. Misconceptions bolster each other and combine to form seductively entrenched *networks of misconception*. Our approach helps to forestall the development of misconception networks by developing a kind of *positive reciprocation*. Because *correctly conceived representations with a high degree of multiple interconnectedness are established*, the fresh entry of fallacious knowledge at any node in the weblike network will fire off so many connections that it would be likely to activate some *misconception-disabling correct knowledge*. Before you can go too far wrong, you are likely to touch something that sets you right.

7. Active Participation, Tutorial Guidance, and Adjunct Support for the Management of Complexity. In an ill-structured domain, knowledge cannot just be handed to the learner. A priori codifications of knowledge are likely to misrepresent. (That is part of what ill-structuredness means.) Hence the importance, increasingly widely recognized today, of active learner involvement in knowledge acquisition, accompanied by opportunistic guidance by expert mentors (which can be incorporated in a computer program--it does not have to be live, one-to-one guidance). Furthermore, aids must be provided to help the learner manage the added complexity that comes with ill-structure. Our hypertext programs allow learners to explore complex conceptual landscapes in many directions, with expert guidance and various kinds of cognitive support (e.g., integrated visual displays). When there are limits to the *explicit* transmission of knowledge, learners will need special kinds of help in figuring things out for themselves. (see [1, 4, 12].)

Recapitulation: A Shift from Single to Multiple Representations and from Generic Schema Retrieval to Situation-Specific Knowledge Assembly

In general, we argue that the goals of advanced knowledge acquisition in complex and ill-structured domains can best be attained (and the problems we have identified avoided) by the development of mental

representations that support *cognitive flexibility*. Central to the cultivation of cognitive flexibility are approaches to learning, instruction, and knowledge representation that: (a) allow an important role for *multiple representations*; (b) view learning as the multidirectional and multiperspectival "criss-crossing" of cases and concepts that make up complex domains' "landscapes" (with resulting interconnectedness along multiple dimensions); and (c) foster the ability to assemble diverse knowledge sources to adaptively fit the needs of a particular knowledge application situation (rather than the search for a precompiled schema that fits the situation). We suggest that theory-based computer hypertext systems can implement the goals and strategies of Cognitive Flexibility Theory, engendering multiple cognitive representations that capture the real-world complexities of the kinds of cases to which abstract conceptual knowledge must be applied.

Acknowledgement

The research reported in this paper was supported by grants from the Army Research Institute (MDA903-86-K-0443), the Office of Naval Research (N00014-88-K-0286, N00014-87-G-0165, N00014-88-K-0077), the Josiah Macy Foundation (B852001), and the Office of Educational Research and Improvement (OEG 0087-C1001). The authors would like to acknowledge the contributions of Jane Adami and Joan Feltovich to various aspects of this research.

References

1. Barrows, H. S., & Tamblyn, R. M. (1980). Problem-based learning. New York: Springer.
2. Burstein, M. (1983). Concept formation by incremental analogical reasoning and debugging. Proceedings of the International Machine Learning Workshop, Champaign, IL.
3. Champagne, A. B., Gunstone, R. F., & Klopfer, L. E. (1985). Effecting changes in cognitive structures among physics students. In L. H. West & A. L. Pines (Eds.), Cognitive structure and conceptual change. Orlando, FL: Academic Press.
4. Collins, A., Brown, J. S., & Newman, S. E. (in press). Cognitive apprenticeship: Teaching the craft of reading, writing, and mathematics. In L. B. Resnick (Ed.), Cognition and instruction: Issues and agenda. Hillsdale, NJ: Erlbaum.
5. Collins, A., & Gentner, D. (in press). How people construct mental models. In S. Quinn & D. Holland (Eds.), Cultural models in thought and language. Cambridge: Cambridge University Press.
6. Coulson, R. L., Feltovich, P. J., & Spiro, R. J. (1986). Foundations of a misunderstanding of the ultrastructural basis of myocardial failure: A reciprocating network of oversimplifications (Tech. Rep. No. 1). Springfield: Southern Illinois University School of Medicine, Conceptual Knowledge Research Project.
7. Feltovich, P. J., Spiro, R. J., & Coulson, R. L. (in press). The nature of conceptual understanding in biomedicine: The deep structure of complex ideas and the development of misconceptions. In D. Evans & V. Patel (Eds.), The cognitive sciences in medicine. Cambridge, MA: M.I.T. Press.
8. Schank, R. C. (1982). Dynamic memory. Cambridge: Cambridge University Press.
9. Spiro, R. J. (1980). Accommodative reconstruction in prose recall. Journal of Verbal Learning and Verbal Behavior, 19, 84-95.
10. Spiro, R. J., Feltovich, P. J., & Coulson, R. L. (in press). Multiple analogies for complex concepts: Antidotes for analogy-induced misconception in advanced knowledge acquisition. In S. Vosniadou & A. Ortony (Eds.), Similarity and analogical reasoning. Cambridge: Cambridge University Press.
11. Spiro, R. J., & Jehng, J. C. (in press). Random access instruction: Theory and technology for the nonlinear and multidimensional traversal of complex subject matter. In D. Nix & R. J. Spiro (Eds.), The Handy Project: New directions in multimedia instruction. Hillsdale, NJ: Erlbaum.
12. Spiro, R. J., Vispoel, W. L., Schmitz, J., Samarapungavan, A., & Boerger, A. (1987). Knowledge acquisition for application: Cognitive flexibility and transfer in complex content domains. In B. C. Britton & S. Glynn (Eds.), Executive control processes. Hillsdale, NJ: Erlbaum.
13. White, B. Y. (1984). Designing computer games to help physics students understand Newton's Law of Motion. Cognition and Instruction, 1, 69-108.
14. White, B. Y., & Frederiksen, J. R. (1987, November). Causal model progressions as a foundation for intelligent learning environments (BBN Report No. 6686). Cambridge, MA: Bolt Beranek and Newman, Inc.
15. Wittgenstein, L. (1953). Philosophical investigations. New York: Macmillan.

A Hybrid Connectionist/Production System Interpretation of Age Differences in Perceptual Learning

Arthur D. Fisk and Wendy A. Rogers

Georgia Institute of Technology

Introduction

In the present paper we will provide a brief discussion of how the appeal to formal models, even used at the conceptual level, can lead to "new directions in cognitive aging research". Following the lead of Salthouse (1988), one of our goals is to encourage efforts to develop and utilize formal models that lead to testable hypothesis concerning age-related cognitive phenomena. Our primary focus is on the explanatory and predictive utility of a specific model for consolidating otherwise confusing age-related practice effects in visual, memory, and hybrid visual/memory search tasks.

Cognitive Aging Research

A review of the literature on differential age-effects in perceptual learning leads to a suggestion of the age independence of search in a consistent mapping (CM) task as well as the age dependence of performance in varied mapping (VM) search tasks[1]. It is important to note that there are two general assertions made in the cognitive aging literature which are relevant to the present discussion. First is the assumption that age-related deficits are due to reductions in controlled processing ability, efficiency of processing, etc. Following this view, VM search effects should differ as a function of age. Second, it is assumed that automatic processes are somehow immune to the deleterious effects of aging. It is true that automatic processes developed prior to senescence may not decline with age, as has been demonstrated in lexical priming studies and the maintenance of Stroop interference effects. However, the assertion that the characteristics of CM search, which lead to the development of automatic processes, do not differ as a function of age has not been agreed upon.

1 Consistently mapped practice involves searching for specific stimuli which always evoke the same response when they occur. For example, in visual search, whenever the target letter 'A' appears in a display it always requires the same response from the subject. When stimuli are variably mapped (VM), i.e., stimuli require responses that constantly change during training, automatic processing does not develop and performance shows little change. For example, in VM training, on one trial the letter 'A' might serve as a target, and therefore require a specified response, while on another trial the letter 'A' would serve as a distractor, and therefore be ignored by the subject. Schneider and Shiffrin (1977) demonstrated that CM practice was a necessary requisite for automatic process development.

In general, previous researchers have inferred that similar CM learning curves early in practice (indexed by a nonsignificant Age X Practice Session interaction) implied that both young and old adults were developing automatic processing at the same rate. The logic of that argument presupposes that all age groups will eventually develop equivalent automatic processing of the CM trained stimuli (e.g., Madden, 1983; Plude & Hoyer, 1981). With very few exceptions (e.g., Salthouse & Somberg, 1982), previous research examining effects of CM training on young and old subjects' performance has provided relatively little practice; that is, the experiments have not been carried out long enough to determine if the automatic processes do actually develop in both age groups.

When extended CM and VM practice is provided to young and old adults, the results are surprising in light of the assumptions existing in the literature. Recently, we (Fisk, McGee, & Giambra, in press; Fisk, Rogers, & Giambra, 1987; Rogers & Fisk, 1988) examined the influence of age on a range of performance measures for consistently mapped and variably mapped visual/memory search tasks. That research extended the previous research examining age-effects in perceptual search tasks by providing extended practice (at least 4,000 CM and 4,000 VM trials), requiring subjects to perform complex semantic-category search tasks, and examining, within-subjects, CM relative to VM performance. In all cases, early in practice, the young and old subjects' performance was similar to that seen in other studies; that is, improving CM search performance for both age groups (measured by reaction time, response time variability, and decreasing comparison slopes) and a nonsignificant Age X Practice Session interaction. However, late in practice, the results suggest that qualitative age differences exist for CM performance; that is, in addition to the fact that older adults are slower (quantitatively different) than young adults, the pattern of performance improvement also differs. Although the older subjects' reaction time decreased with CM practice, there was comparatively little reduction in reaction time, comparison slope, or response variability. These findings hold for simple character search as well as the more complex semantic category search tasks (Fisk & Rogers, 1988). It is also important to note that, with extended practice, the characteristics of the older adults' VM memory scanning and visual search is similar to that of the young subjects. Not only are the patterns of VM search performance equivalent, but the search rate, as measured by the slope of the function relating reaction time to number of comparisons, is also similar (also see Strayer, Wickens, & Braune, 1987).

As noted above, often a plea is made to some amount of reduced mental energy, reduction in controlled processing ability/efficiency, etc. in older adults in order to account for age-related differences in visual/memory search. Given our findings of age-related similarities in VM search (mostly dependent on controlled processing) and age-related differences after extended CM search (indicative of age-related differences in automatic process development), the previous explanations are not appealing. Several questions remain to be answered. Why do young and old adults show equivalent learning early in practice and subsequent divergence late in practice? Why is it that only the young adults show automatic CM target detection? The resolution of these questions must be found in order to explain age-related disruptions, either partial or complete, of automatic process development.

Potential answers to the above questions may be found through the use of a formal model of information processing. The model which seems to account for the apparently discrepant data discussed above is a hybrid connectionist/production system model (Schneider, 1985; Schneider & Detweiler, 1987). The formal modeling approach proposed by Schneider and his colleagues represents a hybrid of two major approaches modeling cognitive mechanisms: connectionist models (for a review see McClelland, Rumelhart, & Hinton, 1986; Schneider, 1987) and production system processing models (e.g., Anderson, 1983). While the utilization of this model for the explanation of empirical data may seem inherently post hoc, the model does make unique and testable predictions concerning age-related effects in visual/memory search. Therefore, we can use this approach not only to consolidate the cognitive aging data already available, but also to drive the direction of future research to further investigate age-related differences.

A Hybrid Connectionist/Production System Model: Conceptually Defined

Schneider's hybrid model proposes two types of learning mechanisms: associative and priority learning. It is assumed that these two different types of learning can occur during CM practice and that both types of learning are necessary in order for the CM trained stimuli to automatically attract attention. Associative learning develops prior to priority learning and is accomplished via a modified Hebb-type learning rule (Stone, 1986) by which an input associatively evokes or retrieves an output message that can be transmitted for additional processing (e.g., "red light" retrieves "press brake pedal"). With associative learning, the learning mechanism changes the connection weights between the input and output information transmissions so that the given input tends to evoke the associated output. Associative learning is responsible for the unitization of memory-sets, categorization (e.g., increasing strength of connections between semantic category exemplars and the higher level category), and increasing connections between target items and required responses. As associative learning develops performance improvements are expected because there is a decrease in the need to pre-load working memory with individual memory set items and the responses associated to those items.

If a deficient associative learning mechanism were the locus of age effects in CM search then old subjects should show no learning during CM search. However, previous studies demonstrate CM learning early in practice for both age groups. Thus the data argue against the associative learning mechanism as the primary source of age-dependent CM search effects. If associative learning is affected to some degree by age, that disruption does not appear sufficient to explain the age differences seen after long-term CM practice.

The other learning mechanism, priority learning, is hypothesized as a mechanism that modifies whether, and how strongly, a given message will be transmitted. Priority learning is an association of a message to a metric of its importance. Messages with high priority are transmitted to higher levels of processing; thus, priority learning provides access of information for follow-on processing (e.g., "red light" - strongly transmits the associated message "press brake" to effect an action). A message will have high or low relevance as a function of its "priority tag". The value of the priority tag is

determined by the number of positive transmissions of the message relative to negative or null events associated with the message transmission (e.g., a match response in a search task [a positive event] relative to a negative controlled processing event [processing with no match and no response]). Each positive event results in the priority tag value being incremented and each negative event results in a decrement of the priority tag. Consistent practice would lead to continual incrementing of the priority tag for target stimuli (when detected) and decrementing of the priority for distractor stimuli. Continued CM practice would lead to a segregation of stimuli such that those stimuli with high priority tags (consistent targets) would become "foreground" and stimuli with very low priority tags (consistent distractors) would become "background". Within the hybrid connectionist model, pure automatic processing (processing without control process assistance) is not possible without sufficient priority learning. A combination of both associative and priority learning allows stimuli to be filtered and messages transmitted without control processing assistance; hence, stimuli can automatically attract attention. If "priority" were maintained by simply increasing or decreasing the associated connection weights (i.e., simply associative learning) then a decrement in "priority" would entail a concurrent loss of associative connections thereby implying a memory loss. However, developing a very low priority tag to exemplars of a CM category such as "four-footed animals" does not imply forgetting the meaning of "cat" or that it is an exemplar of the category.

Given the necessity of the priority learning mechanism for the attainment of automaticity, and the apparent inability of old adults to develop automaticity, it seems logical to focus research efforts on this mechanism in an effort to understand age-related decrements in extended-practice experiments. Note that the disruption of the priority change mechanism need not be all-or-none to yield the age effects we have observed. Indeed, a slowing in the priority change process would, to some extent, lead to a slowing in automatic process development. Therefore, although we have provided old subjects with thousands of trials of practice in an effort to allow the development of automaticity (e.g., Fisk, McGee & Giambra [in press] gave some subjects as many as 12,000 trials), it is still possible that thousands of trials more might be necessary if indeed automatic process development is simply slowed as a function of age. Our results have shown that there is an age-related change in the extended-practice CM performance function which may, at the very least, indicate a disruption in the priority learning mechanism.

Conclusions and Future Directions

The current conceptual framework based on the formal model of Schneider and his colleagues can explain many of the previous results of age-related perceptual learning[2] such as: 1) Some amount of learning for both young and old groups of subjects with consistent practice; 2) Qualitatively equivalent performance early in consistent practice but non-equivalent performance with extended consistent practice; and 3) Maintenance of Stroop interference effects or other previously automatized processes such as lexical

2 Note that this model is derived from young subjects' data and has been found to mimic the practice effects which are typically found for both CM and VM practice.

access. Old subjects' improvement in CM search, relative to VM search, is expected given a relatively intact associative learning mechanism. However, the fact that these subjects demonstrate a failure to achieve automatic processing implies a concurrent inability to sufficiently change the priority tag values. Maintenance of automatic processes developed prior to operational failure in the priority mechanism, e.g., color-word Stroop effects (Cohn, Dustman, & Bradford, 1984), arithmetic stroop effects (Fisk & McGee, 1987), and automatic activation of the lexicon (Cerella & Fozard, 1984; Chiarello, Church, & Williams, 1985) would be unaffected by age.

We are therefore proposing that a failure of the priority tag mechanism may accompany aging while the associative learning mechanism remains intact, or at least is not completely disrupted. Three important predictions may be generated from this conceptualization. These predictions are currently being tested in several experiments. First, the model predicts that old adults should be able to unitize (or develop categories of) stimulus sets without the subsequent development of automatism. Behaviorally, this should result in reduced comparison slopes in the absence of other indices of automatic processing of the stimuli as demonstrated by dual-task or stimulus reversal effects. Another prediction of the model suggests that the strength (i.e., the ability to "attract" attention) of target stimuli relative to distractor stimuli should differ across young and old subjects and lead to differences in the pattern of attention attraction for CM trained stimuli as well as differences in various CM/VM target/distractor reversal effects. As predicted by the model, if the priority change mechanism deteriorates as a function of age, less "strength" difference should develop between CM target and CM distractor stimuli due to a lessening ability to modify the priority of consistent targets relative to the consistent distractors. The third prediction is that well-learned, automatized skills developed prior to senescence should be difficult to modify. Again, this would occur because the, once appropriate, high priority of the stimuli/responses would be less modifiable even in the face of new, consistent, contrary practice. This prediction is being tested by examining the attenuation of Stroop interference effects as a function of age and practice.

Although much of the present review has focussed on visual/memory search, the importance of the model is in its apparent ability to account for a variety of results reported in the cognitive aging literature. More importantly, the adoption of this model has allowed us to make strong, testable predictions about age-related performance under specifiable conditions.

Acknowledgements

Preparation of this paper was supported by a grant from National Institutes of Health No. 1R01AG07654-01 (from National Institute on Aging). We wish to thank L. Barselou and W. Schneider for critical comments on an earlier draft of this paper.

References

Anderson, J. A. (1983). Cognitive and psychological computation with neural models. IEEE Transactions on Systems, Man, and Cybernetics, SMC-13, 799-815.

Cerella, J. & Fozard, J.L. (1984). Lexical access and age. Developmental Psychology, 20, 235-425.

Chiarello, C., Church, K.L., & Williams, W.J. (1985). Automatic and controlled semantic priming: Accuracy, response bias, and aging. Journal of Gerontology, 40, 595-600.

Cohn, N.B., Dustman, R.E., & Bradford, D.C. (1984). Age-related decrements in Stroop color test performance. Journal of Clinical Psychology, 40, 1244-1250.

Fisk, A. D., & McGee, N. D. (1987). Unpublished data.

Fisk, A.D., McGee, N.D. & Giambra, L.M. (in press). Age-related effects on consistent and varied semantic category search performance, Psychology & Aging.

Fisk, A.D., Rogers, W.A., & Giambra, L.M. (1987). Age-related factors affecting automatic process development in visual search: I. Positive and negative responding. Paper presented at the annual meeting of the American Psychological Association, August, New York.

Fisk, A. D., and Rogers, W. A. (1988). Age-dependent influences of memory and visual load in perceptual learning. Paper presented at the Annual Meeting of the American Psychological Association, August, Atlanta.

Madden, D. J. (1983). Aging and distraction by highly familiar stimuli during visual search. Developmental Psychology, 19, 499-507.

McClelland, J. L., Rumelhart, D. E., & Hinton, G. E. (1986). The appeal of parallel distributed processing. In D. E. Rumelhart & J. L. McClelland (Eds.), Parallel distributed processing, Volume 1 (pp 1-44). Cambridge, MA: MIT Press.

Plude, D. J., & Hoyer, W. J. (1981). Adult age differences in visual search as a function of stimulus mapping and processing load. Journal of Gerontology, 36, 598-604.

Rogers, W. A., and Fisk, A. D. (1988). Age-related influences of massed practice, practice type, and practice order on perceptual learning. Paper presented at the Cognitive Aging Conference, April, Atlanta.

Rogers, W. A., & Fisk, A. D. (in press). Age-dependent skills training: Theory, data and guidelines. Proceedings of the Fourth Annual Mid-Central Ergonomics/Human Factors Conference. Champaign, Il: University of Illinois.

Salthouse, T. A. (1988). Initiating the formalization of theories of cognitive aging. Psychology and Aging, 3, 3-16.

Salthouse, T. A., & Somberg, B. L. (1982). Skilled performance: Effects of adult age and experience on elementary processes. Journal of Experimental Psychology: General, 111, 176-207.

Schneider, W. (1985). Toward a model of attention and the development of automaticity. In M. I. Posner & O. S. Martin (Eds.) Attention & Performance XI (pp 475-492). Hillsdale, N.J.: Erlbaum.

Schneider, W. (1987). Connectionism: Is it a paradigm shift for psychology? Behavior Research Methods, Instruments, & Computers, 19, 73-83.

Schneider, W. & Detweiler, M. (in press). A connectionist/ control architecture for working memory. In G. H. Bower (Ed.) The psychology of learning and motivation, Volume 21. New York: Academic Press.

Schneider, W. & Shiffrin, R. M. (1977). Controlled and automatic human information processing: I. Detection, Search, Attention. Psychological Review, 84, 1-66.

Stone, G. O. (1986). An analysis of the delta rule and the learning of statistical associations. In D. E. Rumelhart & J. L. McClelland (Eds.), Parallel distributed processing, Volume 1 (pp 444-459). Cambridge, MA: MIT Press.

Strayer, D. L., Wickens, C. D., & Braune, R. (1987). Adults age differences in the speed and capacity of information processing: 2. An electrophysiological approach. Psychology and Aging, 2, 99-110.

Language Experience and Prose Processing in Adulthood

Roger A. Dixon
University of Victoria

Lars Bäckman
Stockholm Gerontology Research Center

For decades psychologists have administered a variety of cognitive tasks and psychometric intelligence tests to adults of all ages and backgrounds. One goal in this research was to answer such global cognitive developmental questions as: Is there ineluctable, generalized decline in cognitive performance with advancing age? Although exceptions are scattered throughout the literature, one frequently observed result is that, indeed, older adults perform worse than younger adults on numerous complex cognitive tasks. Prior to proffering firm conclusions regarding the generality of aging-related decline, however, cognitive psychologists have recently begun to examine the exceptions to the apparent pattern. In so doing the following arguments are among those that have been noted: (a) Some longitudinal studies have detected substantial individual differences in rate and magnitude of decline; (b) Some cross-sectional studies have reported considerable overlap in distributions, with some older adults performing as well or better than some younger adults on selected tasks; (c) Several investigators have noted that the trajectories of change patterns are not uniform across domains; and (d) Several observers have identified a discrepancy between the apparent everyday (including professional and leisure) competency of many normal older adults and their relatively low observed performance on numerous laboratory cognitive tasks (e.g., Salthouse, 1987). A viable (albeit vague) conclusion may be that, although cognitive aging certainly involves decline in numerous component and complex cognitive processes, there is some evidence for some stabilization for some people in some tasks. A derivative research question then becomes: What are the conditions under which adults might develop or maintain highly skilled levels of performance for cognitively demanding tasks? It is this question that the present brief report addresses.

The specific cognitive activity of interest is reading and recall of prose. In this (as in other) areas there is some evidence that under selected conditions typical age differences (in which young adults perform at a superior level to older adults) may be attenuated for particular performance measures (e.g., Dixon & Bäckman, in press). The conditions considered in the present report are those related to the experience or expertise of older adults. In particular, the interest is in the effects of experience or expertise on the attenuation of age differences in measures of prose reading and recall. It is evident that research on skilled discourse processing and prose recall is not as complete or conclusive as it is for other cognitive skills. This may be due in part to difficulties in providing conceptual constraints on an ambiguous skill, identifying and measuring the performances that indicate that skill, identifying and operationally defining the components, and modeling the connections and patterns of influence. Although a formal model is as yet unavailable, progress in identifying relevant features of the model is summarized.

SKILLED PROSE READING AND RECALL IN ADULTHOOD

There has been little empirical or formal analysis of exceptional prose reading and memory skill. The work that is available ranges from descriptions of legendary performances stemming from the oral tradition of former times to studies of individual performances of contemporary actors or memory specialists. In a previous review (Dixon & Bäckman, in press), we noted that, although cognitive development in adulthood was not a major concern of most of the available studies, in some cases middle-aged and older adults were found to perform at highly skilled levels. In the few studies that have examined prose processing skill and adult age, expertise was conceived of within a normal range of functioning and usually as a variable with which to compare and differentiate within and across adult age groups. For example, after identifying a relevant skill, researchers might divide old and young adult age groups into roughly equivalent high and low skill groups and compare them on selected measures of reading or recall of stories. This literature suggests that there are at least two categories of skills (maintained in adulthood) that could act as cognitive support systems for reading and prose memory in older adults: domain specific (e.g., availability of pertinent prior knowledge or schemata) and domain general (e.g., language experience and verbal skill). Following a brief summary of one study in the domain specific category, we describe recent efforts to examine the influence of the domain general category.

Prior Knowledge

The presence of prior knowledge about the content of text materials could make the reader analogous to the expert in other domains of memory (Ericsson, 1986). The question is whether older adults can, like young adults, use their prior knowledge about the content of passages to provide a semantically rich context in which to encode and retrieve the information actually presented in the text. A subsequent question, of course, is whether successful older adults use the same mechanisms to achieve a high performance level. An additional question is whether under these conditions they perform as well as younger adults, whether experts or novices in the content domain. Of the two studies are relevant to this category, one investigating the effects of prior knowledge and memory for domain-specific stories is summarized (Hultsch & Dixon, 1983). In this study we devised short, structurally equivalent biographical sketches about three entertainment figures for whom young, middle-aged, and older adults possessed significantly more or less prior knowledge than did their counterparts (the figures were Steve Martin (young adults), Susan Hayward (middle-aged adults), and Mary Pickford (old adults)). A fourth biographical sketch of an entertainment figure (Bob Hope) known equivalently by all three age groups was included. The results were encouraging to this line of investigation. Specifically, although there was an overall main effect for age (in favor of young adults), the old adults performed as well as the young adults in remembering what they had read of the stories about Mary Pickford and Bob Hope. Given certain qualifications, this initial manipulation of expertise permits the following interpretation. Age differences in prose recall performance may be present or absent depending on the level of prior knowledge regarding the to-be-remembered material possessed by the various age groups.

Language Experience and Verbal Skill

The second category is that of language experience and verbal skill, both of which may be positively correlated with adult age. It is a typical finding that experience with language and measures of verbal ability (e.g., vocabulary score) increase with advancing age. Initial efforts to examine whether such skills could support text processing in older adults yielded some promising results (e.g., Dixon, Hultsch, Simon, & von Eye, 1984; Meyer & Rice, 1983). In one of these studies (Dixon et al., 1984), we compared high verbal and low verbal younger and older adults on free recall of information from a series of expository texts. There were clear main effects for age (in favor of young adults), verbal skill (in favor of high ability adults), and level of information (where main ideas were recalled better than subordinate ideas). For present purposes, the most pertinent finding was a three-way interaction among age, verbal ability, and level of information. For high verbal subjects (and not low verbal subjects) there were no age differences in recall of the main ideas of the texts, but there were age differences at the level of text details. Thus, high verbal older adults were able to identify and recall the main ideas of these texts as well as young adults, but failed to remember as many of the details.

More recent efforts have been directed at elaborating the operational definition of language experience and at expanding the domain of response measures. Two recent projects are described in somewhat more detail. In the first project, an hypothesis pertaining to whether experience with language and discourse, which increases with normal aging irrespective of professional specialization, could contribute to the development of aging-related cognitive compensatory processing mechanisms. The second project is distinguished from the first in that language experience of a more specific variety is examined in terms of its potential contribution to maintenance of cognitive performance levels. In this project, adults who have had regular experience at remembering, analyzing, and recording the "narratives" of daily life (i.e., diarists) were tested on a series of cognitive tasks.

Metaphoric Processing as Potential Compensatory Component: In a recent chapter, we described a conceptualization of cognitive compensation, especially as it applied to the area of prose reading and recall (Dixon & Bäckman, in press). A brief overview of the logic of this conceptualization follows. Prose reading and recall may be considered molar cognitive tasks, the successful performance of which requires competence in numerous molecular components. Several components of fluent reading and prose recall have been identified (e.g., Baddeley, Logie, Nimmo-Smith, & Brereton, 1985). In normal (young) adults when such components (as indicated by, e.g., working memory span, lexical decision, and letter matching tests) are operating effectively fluent prose reading and recall may result. There is considerable evidence, however, that older adults have impaired abilities in such components. Decrements in these components should have the effect of diminishing performance levels on the molar tasks. As noted above, however, there are some conditions under which older adults seem to have maintained relatively high performance levels for reading and prose recall tasks. This maintenance of molar performance levels could be due to maintenance of performance on the molecular components or to the activation of substitutable molecular components or related skills. In brief, if older adults, suffering decrements in the typical components, employ substitutable molecular components in

processing prose materials then cognitive compensation may be indicated. In a series of studies we examined hypotheses derived from this logic (see Dixon & Bäckman, in press, for an overview).

The first study was designed to investigate whether preliminary evidence for such compensation could be observed in a large sample of unselected young and older adults. Given that there is some evidence for a generalized aging-related increase (or at least maintenance) in metaphoric processing (specifically, metaphor interpretation), we reasoned that, for older adults who perform relatively poorly on typical molecular components, this skill might support performance in reading and remembering metaphoric (but probably not nonmetaphoric) passages. Given certain limitations in the design of the study, a particular pattern of age-related results was required before an inference of compensation could be made (see Dixon & Bäckman, in press). This pattern included the following elements: (a) Positive correlations between measures of molecular components and molar tasks; (b) Significant age differences (young > old adults) on the molecular tasks; (c) Positive correlations (for old adults) between metaphor interpretation and molar tasks; (d) Significant age differences (old > young adults) on metaphor interpretation; (e) Significant age differences (young > old adults on prose recall measures for nonmetaphoric passages); and (f) Significant age differences (old > young adults) or age group equivalence on prose recall measures for metaphoric passages.

Materials pertaining to each of the logical elements (and predictions) were administered to a sample of young (n = 70; **M** age = 25.6 years) and old (n = 66; **M** age = 68.8 years) normal, community-dwelling adults. Of the four structurally equivalent texts, two were metaphoric and two were nonmetaphoric. Participants were asked to read each passage, rate each on a series of dimensions, and then re-tell (in writing) the passages in their own words. Among the response measures for this re-telling of the passages were the following: (a) gist recall, (b) elaborations, or extra-text statements that are consistent with the topic, (c) metaphorical statements not originally in the texts, and (d) macrostatements or summary statements. Component tasks were adapted from those described elsewhere in the literature (e.g., Baddeley et al., 1985). With respect to gist recall, the results matched closely the predicted pattern but, because of two critical variations from the pattern, did not provide even preliminary evidence of compensation via preserved skill in metaphor interpretation. The variations were: (a) although old adults performed better than young adults on metaphor interpretation, their performance on this task was generally uncorrelated to their performance on the prose processing measures, and (b) young adults performed better than old adults on remembering propositions from both metaphoric (**M** = 12.0 vs. 5.0, respectively) and nonmetaphoric (**M** = 13.5 vs. 7.9, respectively) passages. With respect to the other response measures, it was found that for two of them (elaborations and metaphor production) young and old adults performed at equivalent levels. Two qualifications discourage an interpretation of compensation in the case of these important variables: (a) metaphor interpretation was generally uncorrelated with these measures as well, and (b) the frequency of incidences was quite small for both young and old adults.

We are concentrating on two complementary interpretations. First, given that metaphoric skill is not a unitary construct, metaphor interpretation may draw on a different aspect than does metaphoric prose memory. This may account for the fact that the two variables are empirically unrelated. In

another recent study, similar results were found for recall of metaphoric sentences. A second possible interpretation is that the conceptual analysis was overdrawn. The relationship may work, but only for a select sample, not for the unselected sample described above. One avenue presently being pursued is the analysis of individual older adults, each of whom has an unusual constellation of professional and leisure experiences in manipulating and using language (e.g., editors, linguists, and lawyers). As a first step in this direction, data from highly skilled (on metaphor interpretation) older adults from the above study were selected from the sample. Ten subjects constituted this select group of metaphorically skilled older adults. Their mean scores on all text performance measures (four stories by six measures) were compared to those for the remaining old subjects, a similarly select group of young subjects, and the total group of young subjects. The select older subjects performed at a uniformly higher level than the remaining old subjects and in some cases as high as the total group of young subjects. In addition, the correlational pattern was considerably different for the select group of old subjects than for the unselected young and old samples. The correlations between the two domains of variables ranged from -.7 to .7 (as compared to predominantly zero order correlations for the unselected groups). Our tentative conclusion is that further analysis of individual older adults, with extended information on their language experience and on-line measures of reading and recall activities, is in order.

Diary-keeping as Potential Compensatory Experience: In a recent study (Dixon, Fox, & Gould, 1988) we sought to examine a set of language experiences associated with a variety of diary-keeping activities. There were three major objectives. The first goal was to develop measures of experience in diary keeping. Given such measures it would be possible to construct an operational definition of diary-keeping style or (possibly) skill. The second goal was to develop procedures and tasks for measuring a series of cognitive characteristics associated with diary-keeping practice. The third goal was to examine the extent to which diary-keeping experience was associated with (or would support) performance by older adults on selected tasks, arrayed on a continuum from those that were designed to be closely related to diary-keeping activity (near transfer) to those that were not (control or far transfer tasks). A global expectation was that continual practice at remembering, analyzing, and recording one's own daily events (or the narrative of one's daily life) would be related to performance on such tasks as: (a) Production of diary entries for presented hypothetical events, (b) Chronological ordering of randomly presented lists of activities, (c) Recalling lists of daily life events, (d) Recalling narrative stories of daily life and a business dilemma, and (e) Solving life problems presented as diary entries. In this summary we will mention only results pertaining to task (d). It should be noted, then, that two narrative stories presented as diary entries of fictitious protagonists experiencing minor life crises and one structurally equivalent story describing a new business in the midst of an important decision were presented to the participants. Subjects were asked to think aloud as they offered their solutions to the dilemmas in each story. The recall of the stimulus stories was therefore an incidental task. In addition, one control, normed narrative story was presented to each subject with instructions to read and intentionally remember the information presented therein.

Young (n = 15, **M** age = 26.2 years) and old (n = 15, **M** age = 61.9 years) adult nondiarists and young (n = 15, **M** age = 24.5 years) and old (n = 15, **M** age = 60.6 years) adult diarists were tested individually on a series of cognitive tasks. Four measures of diary-keeping experience were developed. The first was a simple measure of duration (in consecutive months) of regular diary keeping experience. Although many of the (nondiarist) respondents had kept a diary for some period of time at one or more points in their lives, diarists (**M** = 205.4 months) had significantly more experience than nondiarists (**M** = 9.4 months). Initial analyses of the prose processing data were concentrated on the age and experience (diarist/nondiarist) factors, where the latter factor was defined in this simple way. Overall, younger adults recalled a greater proportion of propositions from the texts than did older adults. Although several interactions pertaining to other dependent measures were encouraging to the line of investigation, for the text recall measures the old subjects performed similarly, whether they were diarists or not. For example, age differences were attenuated for proportion of propositions recalled of the diary entries (young > old by 4%) as compared to the business problem (young > old by 13%) and the narrative (young > old by 14%). Subsequent analyses concentrated on the adult age and experience factors, where the latter was operationally defined in terms of diary-keeping style (as measured by self-ratings of own diary entries and by independent ratings of produced diary entries). Using a composite measure of diary-keeping style or skill we found that select older diarists were among the highest performing individuals on some of the complex cognitive tasks. In what way would such measures of diary-keeping style or skill be related to the prose processing measures? To investigate this question we computed hierarchical multiple regressions. Entered in the first block for each dependent measure were selected predictor components (or main effects): verbal ability, age, and, as appropriate, the various measures of diary-keeping experience. The interactions (product variables) were entered in the second block. The main results may be summarized as follows. When only the diarists (young and old) were included in the analysis, the composite "skill" variable was not a significant predictor of prose recall performance. When all subjects were included, however, duration of diary keeping and produced diary style (as rated by independent observers) were often significant predictors and, furthermore, interacted with age and verbal ability.

Given the design of this study, it is impossible to disentangle competing potential causal relationships. It appears, however, that there is a recoverable relationship between diary-keeping experience--which is an activity that, at the very least, requires frequent and regular use of language, recall and formulation of naturally occurring and varying narratives, and practice at writing such passages--and cognitive performance. Additional analyses have been directed at the extent to which this relationship is observed in far-transfer, as well as near-transfer, tasks.

SUMMARY

As a field of inquiry, the cognitive psychology of adulthood and aging has undergone numerous conceptual shifts and empirical revelations. To some extent the transitions have run parallel to those apparent in related areas of the cognitive sciences. In this paper we have summarized some recent efforts to understand a particular constellation of mechanisms and influences

on specific examples of complex cognitive functioning. A paramount (and complicating) concern is with changes in the mechanisms, influence patterns, and cognitive performance levels that occur as a function of advancing age. The general issue we have focussed on in this report is the conditions under which normal old adults might maintain skills in reading and memory for prose. We have suggested elsewhere (Dixon & Bäckman, in press) that there are individual differences in (a) the level attained in such demanding cognitive activities, (b) whether the skill is maintained, and (c) the cognitive profile (or reasons for) the maintenance. It is possible that some older adults may rely on the same molecular components (in the performance of molar tasks) as younger adults. It is also possible that some older adults may compensate (either deliberately or automatically) via substitution of alternative, aging-related, cognitively supportive skills (or components thereof).

After summarizing some previous work, we described briefly two sets of studies pertaining to the use of experience (and compensation) in reading and memory for prose. In both it was found that global hypotheses regarding the importance of language-related experience or skill were not supported. Instead, more specific, fine-grained analyses of the components, experience, and the molar skill produced more promising results. We are not prepared to answer the question of whether experience in reading and related activities support skilled levels of performance throughout adulthood. Based on the more recent analyses of the above data, however, we are confident it is a question worthy of continued conceptual and empirical attention.

REFERENCES

Baddeley, A., Logie, R., Nimmo-Smith, I., & Brereton, N. (1985). Components of fluent reading. **Journal of Memory and Language, 24,** 119-131.

Dixon, R.A., & Bäckman, L. (in press). Reading and memory for prose in adulthood: Issues of expertise and compensation. In S.R. Yussen & M.C. Smith (Eds.), **Reading across the life span.** New York: Springer-Verlag.

Dixon, R.A., Fox, D.P., & Gould, O.N. (1988, April). Some cognitive characteristics of young and old adult diarists and nondiarists. Paper presented at the Second Cognitive Aging Conference, Atlanta, GA.

Dixon, R.A., Hultsch, D.F., Simon, E.W., & von Eye, A. (1984). Verbal ability and text structure effects on adult age differences in text recall. **Journal of Verbal Learning and Verbal Behavior, 23,** 569-578.

Ericsson, K.A. (1985). Memory skill. **Canadian Journal of Psychology, 39,** 188-231.

Hultsch, D.F., & Dixon, R.A. (1983). The role of pre-experimental knowledge in text processing in adulthood. **Experimental Aging Research, 9,** 17-22.

Meyer, B.J.F., & Rice, G.E. (1983). Learning and memory from text across the adult life span. In J. Fine & R.O. Freedle (Eds.), **Developmental studies in discourse.** Norwood, NJ: Ablex.

Salthouse, T.A. (1987). Age, experience, and compensation. In C. Schooler & K.W. Schaie (Eds.), **Cognitive functioning and social structure over the life course.** Norwood, NJ: Ablex.

ACKNOWLEDGMENT

The first author acknowledges support for this research from the Natural Sciences and Engineering Research Council of Canada.

EFFECTS OF AGE AND SKILL ON DOMAIN-SPECIFIC VISUAL SEARCH

Stephanie M. Clancy and William J. Hoyer

Syracuse University

It is frequently reported that there is an age-associated decline in visual search and target detection performance. Age-related declines and limitations in a wide variety of capacities and processing resources have been invoked to account for adult age differences in visual information processing performance, such as effortful processing (e.g., Salthouse, in press). However, older adults frequently show preserved function when carrying out attentionally-demanding visual search tasks in familiar domains (e.g., see Charness, 1987; Hoyer, 1985; Salthouse, 1984).

Although cognitive science researchers have studied novice versus expert differences in a variety of domains (e.g., Chi, Feltovich, & Glaser, 1981; Lesgold, Rubinson, Feltovich, Glaser, & Klopfer, in press; Patel & Groen, 1986), relatively little attention has been given to the study of the interaction of age and skill in accounting for differential (or dissociable) patterns of visual information processing performance. We report the results of a study examining skill-related (novice vs expert) and age-related (young vs middle-aged adults) differences in visual search performance under single-task and dual-task conditions. Our primary aim was to examine the extent to which skilled search performance requires less capacity for experts as compared to age-matched controls, and the extent to which there are age-related declines in general search performance. It was expected that younger and middle-aged control subjects would have longer reaction times and higher error rates under dual-task conditions of skilled visual search task compared to age-matched experts, and that older control subjects would be particularly disadvantaged in the dual task condition compared to older skilled subjects. The skill domain selected for study involved target detection in visual displays of microbiological materials.

METHOD

Subjects. Thirty-two female subjects at two age levels (young and middle-aged adults), and at two skill levels (medical technologists and control group) were tested individually in two one-hour sessions. The young adults in the skilled group had a mean age = 26.5 years (range = 23-29 years). The mean age of the middle-aged skilled individuals was 41.6 years (range 31-56 years). In the control group, the mean age of the younger subjects was 21.6 years (range 20-26 years). Middle-aged control subjects had a mean age = 47.0 years (range 37-58 years). Scores on Lauzon's (1987) Clinical Microbiology Laboratory Test for the medical technologists ranged from 70% to 100% compared to 20% to 40% for the control subjects. All subjects were in good to excellent health (based on self-report) and had visual acuity of 20/40 or better (with correction) as measured by an Orthorater.

Design and materials. We used a dual-task procedure to examine age and skill

differences in the attentional demands of visual search (e.g., see Logan, 1978; Madden, 1987). In the dual task condition, the secondary task was a simple tone reaction time, and the primary task was a complex visual search reaction time task. The visual search task required the subject to make a decision about the presence or absence of a target or probe item in a previously presented display. Two types of visual search tasks (skilled and general) were administered using computer-driven tachistoscopic displays to all subjects under single-task and dual-task conditions. The stimulus materials for skilled visual search task consisted of color slides (35 mm) of gram stains. The samples used as stimuli reflected pathogens commonly encountered in the microbiology section of a clinical pathology laboratory. Two blocks of forty-eight pairs of field displays and probes were presented. Each of the positive probes had some degree of diagnostic significance, and each was an exact copy of part of the field display. The stimulus materials for general visual search consisted of two blocks of forty-eight pairs of field and probe display slides. Stimuli varied along the dimensions of shading, size, shape, number of features, and orientation. The search displays consisted of four, six or eight objects.

Procedure. The procedures for both visual search tasks were as follows: 1) A LED warning light was presented for 100 ms at the central fixation point. 2) The visual display to be searched was then projected for 500 ms. 3) A mask of black and red diagonal lines was displayed for 100 ms, followed immediately by a centrally-presented probe. 4) A yes/no response indicating the presence or absence of the probe in the previous visual display removed the probe. 5) Onset of the LED for the next trial followed after a variable intertrial interval averaging 1.75 sec (range 1.25-2.25 sec). For each search task and block of dual-task trials, there were 30 trials in which both tone and visual display were presented and 18 trials in which only a visual display was presented. During the dual-task trials, tone onset followed the visual display by either 100, 550 or 700 ms. The 30 tone-present trials contained 10 trials at each SOA value. The order of the dual-task and single-task trials was randomized within each block of trials and the presentation of task type was counterbalanced across subjects. Viewing conditions were binocular at approximately 55 cd/m projected luminance.

RESULTS

For both skilled and general visual search, median correct reaction times, proportional tone reaction times for correct trials, and error rates were calculated for each individual. Separate split-plot ANOVAs were computed for each measure with age and skill as between-subjects factors, and probe type and trial type (tone present vs. tone absent) as within-subjects factors. The simple RT data for skilled visual search are displayed in Figure 1. A significant main effect of age, F (1,28) = 14.90, MSe = 5921, revealed that middle-aged subjects exhibited longer reaction times (M = 1460.31 ms, SD = 70.08) than younger subjects (M = 938.10 ms, SD = 22.20). A significant age by skill by trial type (i.e., tone present vs. tone absent) interaction was also observed, F (1,28) = 5.82, MSe = 1612. A test of simple effects revealed that in the middle-aged group, the skill by trial type interaction was significant, F (1,14) = 4.62, MSe =3086. As shown in Figure 1, the middle-aged control subjects were significantly slower (M = 1794.10 ms, SD =

24.72) during the tone present trials than the middle-aged medical technologists (M = 1267.50 ms, SD = 26.59).

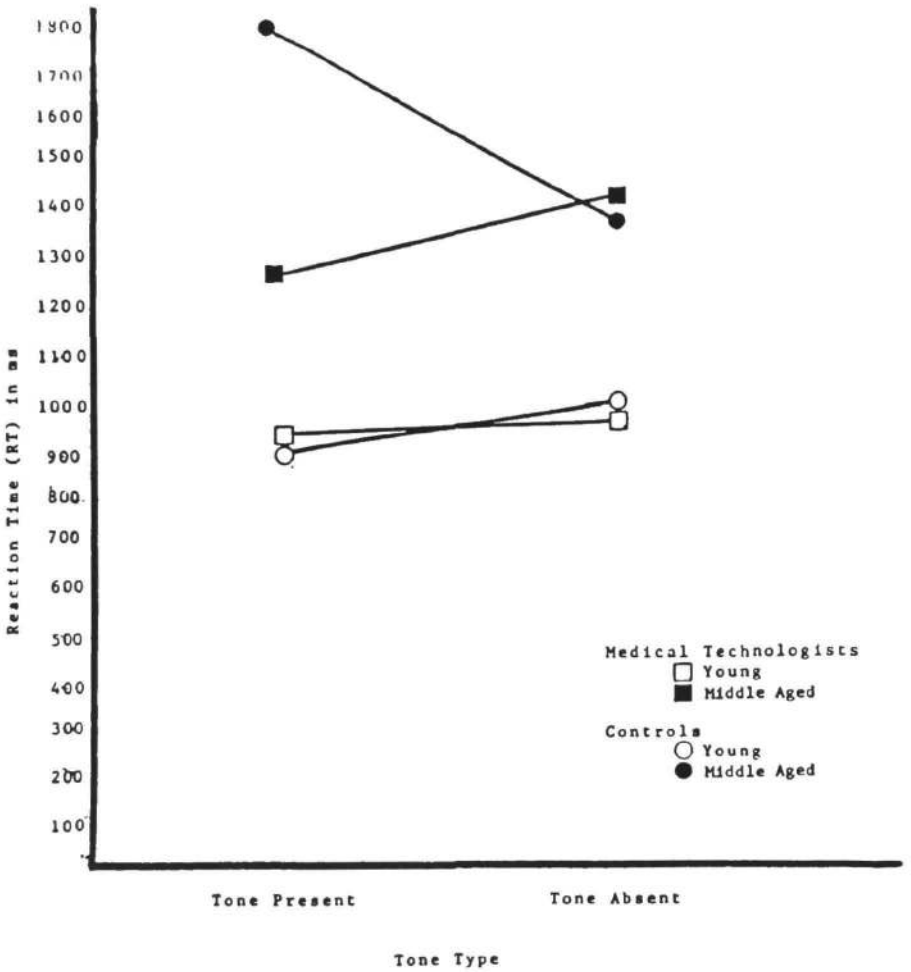


Figure 1. Mean Skilled Visual Search Reaction Time (RT) in ms as a Function of Tone, Skill and Age in Experiment II.

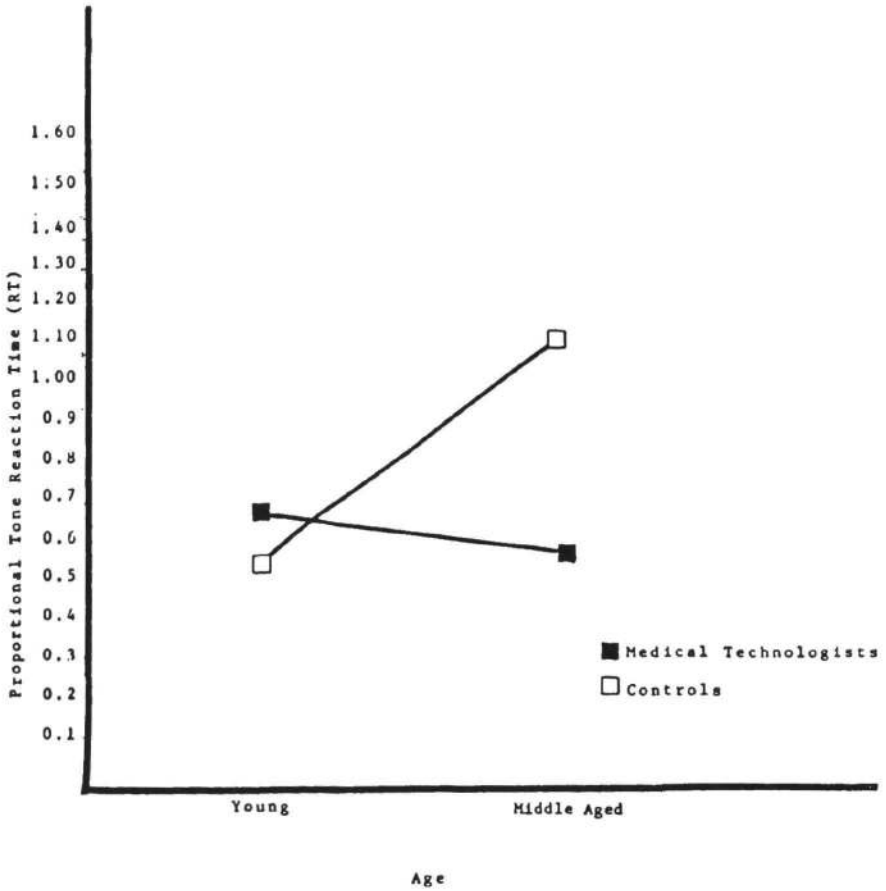


Figure 2. Proportional Tone Reaction Time (RT) as a Function of Age and Skill in Experiment II Under Skilled Visual Search Conditions.

Proportional tone reaction time data are displayed in Figure 2. Analyses revealed a significant main effect of age, F (1,28) = 7.49, MSe = .51 with middle-aged adults having higher proportional tones (M = .80, SD = .57) than younger subjects (M = .52, SD = .33). In addition, a main effect of probe type F (1,28) = 5.07, MSe = .14 was also obtained in which the proportional reaction time to tones during negative probe trials (M = .72, SD = .54) was higher than proportional reaction time to tones during positive probe trials (M = .60, SD = .42). Analyses also revealed an age by skill interaction, F (1,28) = 10.41, MSe = .51. No age effects were observed in proportional tone in the medical technology group, F (1,14) = .32, MSe = .24. However, in the control group, middle-aged subjects had significantly larger proportional tone reaction times (M = 1.07, SD = .67) than younger individuals (M = .45, SD = .28), F (1,14) = 11.70, MSe = .77.

For general visual search, no main effects of skill or interaction effects involving skill were expected; nor were they obtained. By extending the results of previous studies of older adults (e.g., see Madden, 1986), the dual task condition was expected to be relatively more difficult for middle-aged subjects than for younger adults. As predicted, there was a main effect of age, F (1,29) = 11.92, with middle-aged subjects having longer reaction times (M = 1516.60 ms, SD = 53.13) than younger subjects (M = 1102.10 ms, SD = 27.52). Although the age by dual task interaction did not reach significance based on the analysis of simple RT data, the analysis of proportional tone RT did yield a statistically significant age by dual task effect. Simple effects and other interactions based on the analysis of proportional reaction times and errors supported the expected finding of age-related decline as processing demands increased.

DISCUSSION

It is interesting that age differences in the efficiency of general visual search can be found even when comparing middle-aged subjects with young adults. Our results raise several important questions regarding nature of age-related decline in domain-general visual processing in the absence of age-related reductions in domain-specific visual search. There are several ways of interpreting the beneficial effects of the observer's domain-specific experience on adult age differences in search performance. First, consistent with previous work, domain-general computational processes are susceptible to age-related decline because they require more of some unspecified limited capacity, but once developed less of this general capacity is required for the "run off" or execution of performance (Salthouse, 1987). Although it may be reasonable to speculate that with advancing age, there is increased constraint on the extent to which general computational processes can be applied to specific contents, it is difficult to formalize such an explanation.

Alternatively, it can be suggested that if skilled observers know what to see or look for in familiar displays, then compared to less skilled observers, less processing time (and/or capacity) is required for handling nonsalient information. That is, compared to novices, skilled observers use different (or fewer) processing components to perform the same molar task. Thus, age-normal deficits in processing speed or capacity are either circumvented or such processes as orientation, recognition, input analysis, filtering, search, comparison, and other normally age-sensitive, capacity-demanding processes have sufficient lead time to "run off" without noticeable deficit. For example, there may be reduced susceptibility to distractor information in skilled domains. Compared to younger adults, older adults have been shown to be more susceptable to the effects of noise factors which disturb the reception and processing of incoming signals (e.g., see Lindholm & Parkinson, 1983; Sekuler & Ball, 1986). Skilled older observers, may be able to effectively "see through" noise in familiar information processing tasks, even though older adults are generally more prone to noise distractors when searching unfamiliar domains/displays. It may be that the disadvantageous effects of noise are at minimum for experts, within a relatively wide age range, because of the benefits of attentional selectivity and preparation, which attenuate the typical pattern of age-related

differences found when familiarity or preparation effects are not available (e.g., see Hoyer & Familant, 1987). That is, domain-specific search minimizes the differential effects of distractors on age differences in visual search.

It is also reasonable to suggest that search performance in skilled domains involves a criterion adjustment at the encoding or identification stages of information processing. Familiar context may affect the observer's criterion for accepting partial stimulus information as evidence for the presence or absence of a probe. Particular features of the display may operate as primes for the expert, which facilitate the encoding and stimulus identification processes through network activation (e.g., Chiarello, Church, & Hoyer, 1985). Thus, skilled visual search performance may involve increased contextualization and the activation or imposition of useful constraints on the processing of visual information. Although it may be useful to explain age by skill interactions in terms of contextual factors which support access and/or retrieval of information within domains of expertise, the problem remains of critically testing the various available explanations of the paradoxical effects of age and skill in visual search performance.

Acknowledgements: This research was supported by NIA research grant AG 06041 to WJH.

References

Charness, N. (1987). Age and expertise: Responding to Talland's challenge. In L. W. Poon, D. C. Rubin, & B. A. Wilson (Eds.), Cognition in everyday life: Proceedings of the George A. Talland Memorial Conference on Aging and Memory.

Chi, M. T. H., Feltovich, P., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. Cognitive Science, 5, 121-152.

Chiarello, C., Church, K., & Hoyer, W. J. (1985). Automatic and controlled semantic priming: Accuracy, response bias, and aging. Journal of Gerontology, 40, 593-600.

Hoyer, W. J. (1985). Aging and the development of expert cognition. In T.M. Schlechter, & M. P. Toglia (Eds.), New Directions in Cognitive Science (pp. 69-87). Norwood, NJ: Ablex.

Hoyer, W. J., & Familant, M. E. (1987). Adult age differences in the rate of processing expectancy information. Cognitive Development, 2, 57-70.

Lauzon, R. (1987). A clinical pathology laboratory knowledge test. Unpublished test. Syracuse, NY: State University of New York Health Sciences Center.

Lesgold, A. M., Rubinson, H., Feltovich, P. J., Glaser, R., & Klopfer, D. (in press). Expertise in complex skills: Diagnosing x-ray pictures. In M. Chi, R. Glaser, & M. Farr (Eds.), The nature of expertise.

Lindholm, J. M., & Parkinson, S. R. (1983). An interpretation of age-related differences in letter-matching performance. Perception and Psychophysics, 33, 283-294.

Logan, G. D. (1978). Attention demands of visual search. Memory and Cognition, 6, 446-453.

Madden, D. J. (1986). Adult age differences in the attentional capacity demands of visual search. Cognitive Development, 1, 335-363.

Patel, V. L., & Groen, G. J. (1986). Knowledge-based solution strategies in medical reasoning. Cognitive Science, 10, 91-116.

Salthouse, T. A. (1984). Effects of age and skill in typing. Journal Experimental Psychology: General, 113, 345-371.

Salthouse, T.A. (1988). Initiating the formalization of theories of cognitive aging. Psychology and Aging, 3, 3-16.

Salthouse, T. A. (in press). The role of processing resources in cognitive aging. In M. L Howe & C. J. Brainerd (Eds.), Cognitive development in adulthood. New York: Springer-Verlag.

PATCHING UP OLD PLANS

Katia Sycara

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213)

ABSTRACT

Recent research has demonstrated the value of re-using old plans rather than creating plans from scratch. This approach to planning creates the need for efficient and flexible plan adaptation methods to transform a past plan to fit the current problem. A characteristic of plans is that they often fail. This creates the need for efficient and flexible plan repair methods. We propose a uniform treatment of the two issues, plan adaptation and repair, based on a combination of Case-Base Reasoning and heuristics. Plan adaptation involves incremental modification of the old plan, and fixing of anticipated problems through similarity-based retrieval of cases that supply appropriate modifications. Plan repair involves explanation-based retrieval of previous failures that supply possible repairs. A selected repair is then adapted to fit the current failure. The proposed approach gives a planner the flexibility to access a broad range of adaptation and repair strategies not available to planners that use either of the two methods in isolation. The approach has been implemented in the PERSUADER, a case-based planner that generates and repairs plans to resolve labor management disputes.

INTRODUCTION

Recent research has demonstrated the value of re-using old plans rather than generating plans from scratch [Alterman 87, Hammond 86, Simpson 85, Kolodner et al. 85, Turner 87]. If a planner wants to re-use plans effectively it must have ways of *adapting* a previous plan to fit the current situation. Re-using plans is especially useful in domains that are ill-defined and have no strong causal theories or well-understood empirical regularities. One consequence of these characteristics is that solutions may fail and may need *repair*. This paper addresses the issues of plan adaptation and repair. The proposed approach involves the integration of Case-Based Reasoning (CBR) and heuristics.

Adaptation in the past has been done using only heuristics (e.g., [Hammond 86]). We are using CBR in a novel way: to suggest means of adaptation. The proposed process of coming up with an initial solution is an incremental process of adjusting the previous plan using CBR and heuristics, and anticipating possible failures by examining the case memory for failures that have the same features as the features in the current problem. Using CBR for adaptation adds flexibility since, as more experiences are acquired, the planner has an increasingly larger repertoire of adaptations that it can choose from. In planning for the satisfaction of multiple goals, the precedent plan selected might not make provisions for all input goals. CBR provides means of adapting a plan that satisfies some of the goals, to satisfy the rest. In addition, since cases incorporate accumulated expertise and changing circumstances, the proposed adaptations are more closely suited to a current problem.

After a plan has been created, it is tried out in the world. If the plan fails, it needs to be repaired. Our approach to repair uses an explanation for the failure provided by environmental feedback. This explanation is used to index into the case memory to retrieve and adapt the repair of previous similar failures to the repair needs of the current situation. In other words, Case-Based Reasoning in the space of failures is employed.

---

[1]This research was funded in part by the Army Research Office under contract No. DAAG 29-85-K-00230.

The spectrum of situations where plans will fail cannot be anticipated. Thus, use of heuristics or hardwired TOPs alone [Hammond 86, Turner 87] do not provide a flexible enough methodology for dealing with failed plans. CBR is more flexible, since, as memory is enriched with new experiences, new ways that plans can fail and new repairs become available. Simpson [Simpson 85] also has used CBR for failure recovery and repair. His approach differs from ours in that he uses similarity-based retrieval of failed cases. Hammond [Hammond 86] finds an explanation of a plan failure via a set of causal rules that describe effects of actions under different circumstances. The explanation is then used to find a planning TOP with repair strategies. In our model, the explanation is provided via environmental feedback. This is realistic for problem domains with incomplete and changing knowledge and no strong domain model. For such problems, automatic failure explanation methods are not applicable. The features of the explanation are used as indices to retrieve previous failures and access the associated repairs.

Our approach is embedded in the computer program PERSUADER, which, acting as a labor mediator, resolves conjunctive conflicting goals of a union and company by finding compromises acceptable to them. Although the program operates in the domain of labor relations, the techniques it uses are domain independent.

THE PERSUADER

The PERSUADER is a planner that integrates Case-Based Reasoning with Preference Analysis (a planning method based on multi-attribute decision theory [Sycara 87, Sycara 88]) to create acceptable compromises in labor management contract negotiations. The PERSUADER acts as a labor mediator. Its task is to help the disputants arrive at a mutually acceptable compromise plan (contract). To reach its goal, the PERSUADER must plan actions to achieve instrumental reductions in the difference between the parties' positions. Planning must be iterative and reactive because knowledge is incomplete (e.g., models of the agents' intentions must be inferred), and dynamically changing (success is met by shifts in positions which necessitates further planning).

The PERSUADER's input is the set of conflicting goals (e.g., wages, pensions, holidays, seniority language, management rights) of the company and its local union, and the context of the dispute (e.g., economic conditions in the industry, general economic conditions, information about the disputants). Its final output is either an agreed upon compromise, or an indication of inability to solve the problem if the parties to the dispute did not reach agreement within a particular number of proposals (to simulate the inability of the parties in the real world to reach agreement before a strike deadline). To perform its tasks, the PERSUADER uses knowledge of past negotiations and settlements (cases), knowledge of the labor domain, and commonsense knowledge of human goals and behavior. Cases are indexed in memory via salient domain features, such as the industry to which the company belongs, the international union to which the local belongs, the geographical location of the company, the union members' job classification, and the final contract.[2]In labor mediation, the most important feature is the industry to which the company belongs. Hence, if a current case involving a competitor is in the set of retrieved cases, it is the one used as a basis for reasoning. The concept of a negotiation failure is an *impasse*, a situation where the planner has proposed a compromise that has been rejected by at least one agent. A list of impasses represent the negotiation process. In addition to salient domain features, impasses are indexed by the failure reason and objectionable negotiation issue. Associated with impasses are plans used to repair them.

To construct an initial compromise, the PERSUADER (a) retrieves appropriate precedent cases from memory using the features as memory probes, (b) selects the most appropriate case(s) from those retrieved using an evaluation function based on a prioritization of the features, (c) accesses the plan used in the selected case, and (d) adapts the precedent plan to fit the current situation.

Once a proposal has been generated, the PERSUADER proposes it to the parties. If the parties agree, the case memory is updated with a successful episode. If at least one party disagrees, the PERSUADER iteratively performs the following tasks: it generate persuasive arguments to change the evaluation of the

[2]The PERSUADER's memory is based on *generalized episodes* [Kolodner et al. 85].

objecting party with respect to the proposed compromise, or incrementally repairs the rejected compromise to increase its acceptability.

The PERSUADER also stores failures along with the reason for failure (if one can be found) as well as dependencies among decisions taken at different times of the negotiation. Failures are stored so that they can be recalled in situations with similar features to the one where the failure occurred, thus warning the problem solver about potential problems, as well as providing appropriate repairs. Unlike other case-based planners (e.g. [Hammond 86]) that only avoid problems that they can anticipate at the beginning of planning, in the PERSUADER, warning/avoidance of problems occurs at each decision point.

PLAN ADAPTATION

The plan adaptation process consists of the following steps:

- Adjust the accessed precedent plan to compensate for its dissimilarities from the "ideal" precedent for the current problem
- Modify the adjusted plan to take into account detailed information about the current problem
- Anticipate failures of the modified plan and fix it appropriately

Although we present the steps involved in plan adaptation as sequential, a reasoner may go through many iterations of the loop in order to construct a suitable plan. This happens because a contemplated modification may interact with the already existing part of the plan. If this interaction cannot be fixed, another case may need to be retrieved to suggest a more suitable modification.

Precedent Adjustment

The ideal precedent plan is one that was used in a case whose indices have the same values as in the current case. In labor mediation, for construction of an initial proposal, the ideal precedent is a case where the disputant company is a competitor of the current disputant company, the geographical location of the companies is the same, the local unions belong to the same international union, the job classification of the union members is the same, and the two companies have the same size and organizational structure. Since a retrieved precedent case usually differs from the ideal, the plan used has to be adjusted. For example, if a precedent case from a similar industry (not a competitor) is available, then an adjustment of the values of wages has to be made to compensate for dissimilarities along the industry dimension. In labor mediation, this adjustment is called the industry differential. If the company in the precedent case is in a different geographical location than the company involved in the current dispute, then the proposed wages have to be adjusted by the difference in the cost of living in the two locations. This is called the area differential.

Precedent adjustment is done using a library of heuristics. The difference between the current precedent and the ideal along the important problem dimensions (e.g., industry, geographical location) is used to index into the library of heuristics to find the one to apply.

Consider, for example, the PERSUADER trying to find a compromise for Thompson Inc, a company producing airplane frames and its union. The union wants 15% wage increase, 7% increase in pensions, no subcontracting, and strict seniority governing promotions and layoffs. The company wants no wage increase, no pension increase, unlimited subcontracting, and no seniority (i.e., promotions and layoffs to be determined by criteria chosen by the company). The PERSUADER searches memory for similar past contracts. It cannot find contracts of competitors but finds contracts of similar industries. Out of those, it selects the contract of Baker Inc. company since its product (car chassis) is most similar to the products of Thompson Inc., and since its location (Oregon, a northwestern state) is similar to Thompson's (Northern California). The Baker Inc. contract provided 10% wage increase, 4% pension increase and subcontracting for limited time periods. Since the Baker Inc. contract makes no provisions for promotions and layoffs, another contract making such provisions is sought. The contract of Schmidt Inc., a company that makes steel cases is selected out of those retrieved. It stipulates that in layoffs and promotions the company should observe the principle of seniority in conjunction with the worker's ability

to perform the work.

The information from the Baker Inc. and Schmidt Inc. contracts is combined to form a candidate contract for Thompson Inc. This contract is adjusted by applying to it industry and area differentials between Oregon and California. These adjustments result in 12% wage increase, 5% pension increase, subcontracting for limited time periods, and seniority and ability as co-determinants for promotions and layoffs.

The result of precedent adjustment is a base-line plan, called the "ballpark" plan. A ballpark plan is the best that can be constructed without taking details into account. The ballpark plan is modified further taking into consideration important particulars of the current problem.

Modification of the Ballpark Plan

After constructing a ballpark plan, it has to be evaluated for appropriateness to the current case. There are three categories of knowledge that the PERSUADER takes into consideration when criticizing a ballpark plan:

- Knowledge of unacceptability conditions
- More detailed knowledge of the situation of the disputants
- Knowledge of the dispute context and its effects on the situation

During evaluation, critics associated with these knowledge sources are activated. These critics are prioritized and considered in order of importance. A critic is most important if failure to apply it would result with greater probability in plan unacceptability. For example, a check is always made to see whether the company will be able to afford the ballpark economic package. This check is important since, if a company cannot afford the economic package, it most probably will reject the proposed settlement. If it is found that the company can afford the economic package, then critics associated with possible states of the company finances are applied. Such financial considerations include whether the company has suffered losses in the recent past, and whether the company has traditionally paid above, below, or industry average. A set of critics associated with the context of the dispute is then applied. In labor mediation context knowledge is almost entirely economic. Such knowledge includes considerations for the whole economy (recession, inflation etc), economic conditions of the industry to which the disputant company belongs, economic conditions of the geographical location of the company, and labor supply in the area.

If the application of a critic suggests that the plan needs further modification, salient features associated with the critics are used to search memory. The evaluation function is used to select the best out of the retrieved cases. The plan used in that case is checked for applicability. If the considered plan's preconditions match the current situation, then the plan is applied. If not, either the plan is modified using heuristics, or the next best case from those retrieved is considered.

Case-Based Reasoning is the preferred modification method in the PERSUADER. The rationale is that a previous case best reflects the interactions present in the situation. Moreover, the fact that the suggested modification has worked in a similar situation can be used as justification more convincing to the agents than invocation of a rule. If previous cases are not available, then rules associated with the critics are used for modification.

In trying to modify the ballpark plan for Thompson Inc., the PERSUADER finds out that the company cannot afford the contemplated economic package. The case memory is now searched for cases where the same situation (inability of the company to afford the economic package) had occurred for similar companies. The case of Ironside Inc., a company making truck frames, is selected out of the set of retrieved cases as most similar because of similarity of the product, same geographical area, and similarity of the issues involved in the negotiation. In that case, the plan "pass the extra cost to the consumer" had been applied. A precondition of this plan is that the market for the product is not sensitive to price increases. Since the precondition is satisfied in the case of airplane frames, the plan is applied (i.e. the contemplated contract is not changed).

Searching memory with index INABILITY-TO-PAY and ECON-PKGE
3 cases found... Select case3,
since it is similar product, same area, same issues in economic package
Looking at the plan "pass the extra cost to the consumer" from case3
Check preconditions of plan used in this case

Since the market is insensitive to product price change
for airplane frames, plan applicable
Apply plan used in case3 to economic package

The PERSUADER now finds that Thompson Inc. has had losses of 3% in recent years. The case memory is now searched for similar situations. Since no pertinent cases are found, the PERSUADER uses a heuristic associated with the condition of recent losses that supplies the advice to reduce the contemplated increases for wages and pensions by half of the percentage losses. This results in 10.5% wage increase and 3.5% increase in pensions.

Failure Anticipation

Before proposing the generated compromise, the PERSUADER checks to see whether the solution might engender some unforeseen problems. The knowledge that a plan has failed in the past can suggest to the planner the potential for failure if the plan is adopted in the current situation. In contrast to other case-based systems (e.g., [Hammond 86]) that anticipate problems *before* constructing a plan, the PERSUADER anticipates failures as the *final step* in plan construction. This is appropriate in domains where the sought after plan is a compromise plan and thus, it is not apparent at the beginning of planning what exact form this compromise will take.

Failure anticipation is done through *intentional reminding* [Schank 82] of failures. This is appropriate since the PERSUADER does not assume a strong domain model that could be used to index failures only via the features that contributed to the failure. The conjunction of the plan's features and the index "FAILURE" are used as a memory probe. This probe returns cases where the contemplated plan has failed. The most similar case is selected and the repair used in that case (if present) is accessed. The process (knowledge extraction, evaluation) is applied to the selected repair to yield an appropriate adaptation to avoid the potential problem.

Having the repair stored in memory is best. This is not always possible. Often, it may be known that a plan was inadequate but no explanation or repair was found. In this case, the knowledge of a past failure can still warn the planner of the presence of a potential failure.

Before proposing the updated compromise to the Thompson Inc. labor dispute, the PERSUADER searches memory to discover potential problems. It finds problems with the contemplated subcontracting language. It retrieves a case where the union had filed a grievance protesting that the company resorted to subcontracting instead of recalling laid off workers. The arbitrator in that case did not vindicate the union since the subcontracting clause restricted the company only as to the duration of subcontracting. The arbitrator, however, mentioned in the award the language needed to safeguard the union against such practice. The language was "The company has the right to subcontract for limited periods of time, and when it is clear that no work will be lost to union members".

Searching memory with index FAILURE, SUBCONT-LANG, LIMITED-TIME
1 case found...
Looking at the modification "no work lost to union members" from case1

No precondition to check
Apply modification used in case1 to subcontract language

PLAN REPAIR

It is a known fact of life that plans fail and need repair. In labor mediation, failure of a proposed compromise means that an agent has rejected it. Repair is needed to improve the acceptability of the

rejected plan. The PERSUADER has two ways of reacting to plan failure/rejection: changing the rejecting agent's evaluation of the plan through persuasive argumentation, and modifying/repairing the plan so that it will be more acceptable. Persuasive argumentation is tried first, since, if the objecting agents can be convinced to change their utilities and accept the compromise, then a successful resolution has been found. If, on the other hand, a rejected compromise is modified/repaired, the repair may make it objectionable to agents that had agreed before. Thus, only after persuasive argumentation is no longer judged effective by the planner (i.e. all applicable arguments that the planner could generate have been tried and rejected), is repair tried. For details of generating persuasive arguments see [Sycara 87, Sycara 85a, Sycara 85b].

When a failed compromise plan needs to be improved, the PERSUADER ascertains from the rejecting agent's feedback the objectionable goals, the reason for the rejection and the importance the agent attaches to the goals. Each objectionable goal/issue and reason are used as probes to select impasses with the same stated impasse goal and impasse cause as in the present failure. In other words, CBR is employed in the space of impasses. The selected impasse supplies repairs that will hopefully improve the rejected solution. If no appropriate impasses can be found, the PERSUADER uses standard heuristics that it knows about.

In multivariate planning there are many ways a plan could be modified/repaired. A planner seeks not only a plausible repair but one that with some confidence *improves* the rejected plan. After a repair is applied, the resulting compromise is evaluated using the parties' satisfaction with the compromise.[3]The criterion of plan improvement that the PERSUADER uses is whether the contemplated repair increases the rejecting agent's satisfaction more than it might decrease the satisfaction of the agent(s) who have agreed to the compromise. Without an ability to predict which repair has a chance of being accepted, the planner could propose repairs that do not converge to a mutually acceptable compromise.

The PERSUADER's strategy for repair is explanation-based where the explanation (reason for rejection) is supplied by the rejecting agent. This is realistic for complex domains where there is no strong domain model, and plan failure depends in part on idiosyncratic behavior of the agents. In such domains, automatic failure explanation methods are not applicable.

When presented with the compromise resulting from the adaptation process, Thompson Inc. objects to the seniority language saying that, since many of its key employees are junior, they will be the first to be laid off. Having this explanation the PERSUADER retrieves a case where the same objection was raised. The seniority language in that case was amended to read: "Key employees will be excepted from the seniority rule for layoffs; the company will designate whom it considers key employees".

Searching memory with index FAILURE, SENIORITY, KEY-EMPLOYEES
2 impasses found... Select impasse1,
since it is same industry, same area, same job classification
Looking at the repair "except key employees" from impasse1

No precondition to check
Apply repair used in impasse1 to seniority language

SUMMARY AND CONCLUSIONS

We have advocated the integration of CBR and heuristics for plan adaptation and repair. Such integration gives a planner the flexibility to consider a broad range of adaptation and repair strategies that planners that use either of the two methods in isolation cannot have access to. The use of CBR allows the planner to check and select strategies that do not introduce additional problems. Since cases incorporate accumulated expertise and changing circumstances, the proposed adaptations and repairs are closely suited to a current problem. Repairing plans using CBR is particularly suitable for domains with incomplete and changing knowledge and no strong domain model. These characteristics typify most "real

[3]An agent's satisfaction with a compromise, called his payoff, is calculated using a method based on multi-attribute utility theory. For details of this, see [Sycara 87, Sycara 88].

world" domains. In addition, a planner has the flexibility to try multiple strategies suggested by cases and choose the most appropriate. Integration of CBR and heuristics for adaptation and repair gives a planner the choice of using knowledge in either form: generalized (inferential rules) or specific (cases).

REFERENCES

[Alterman 87] Alterman, R.
The Operational Level of a Commonsense Planner.
In *The Ninth Annual Conference of the Cognitive Science Society*, pages 622-631. Seattle, WA., 1987.

[Hammond 86] Hammond, K.J.
CHEF: A model of case-based planning.
In *AAAI-86*, pages 267-271. AAAI, Philadelphia, PA, 1986.

[Kolodner et al. 85]
Kolodner, J.L., Simpson, R.L., and Sycara-Cyranski, K.
A process model of case-based reasoning in problem solving.
In *IJCAI-85*, pages 284-290. IJCAI, Los Angeles, CA, 1985.

[Schank 82] Schank, R.C.
Dynamic Memory.
Cambridge University Press, Cambridge, 1982.

[Simpson 85] Simpson, R.L.
A Computer Model of Case-Based Reasoning in Problem Solving: An Investigation in the Domain of Dispute Mediation.
PhD thesis, School of Information and Computer Science Georgia Institute of Technology, 1985.

[Sycara 85a] Sycara-Cyranski, K.
Arguments of persuasion in labor mediation.
In *IJCAI-85*, pages 294-296. IJCAI, Los Angeles, CA, 1985.

[Sycara 85b] Sycara-Cyranski, K.
Persuasive argumentation in resolution of collective bargaining impasses.
In *Seventh Annual Conference of the Cognitive Science Society*, pages 356-360. Cognitive Science Society, Irvine, CA, 1985.

[Sycara 87] Sycara, K.
Resolving Adversarial Conflicts: An Approach Integrating Case-Based and Analytic Methods.
PhD thesis, School of Information and Computer Science Georgia Institute of Technology, 1987.

[Sycara 88] Sycara, K.
Utility Theory in Conflict Resolution.
Annals of Operations Research , 1988.
(to appear).

[Turner 87] Turner, R.
Modifying Previously-Used Plans to Fit New Situations.
In *The Ninth Annual Conference of the Cognitive Science Society*. Seattle, WA., 1987.

Systematicity as a Selection Constraint in Analogical Mapping

Catherine Clement and Dedre Gentner
Psychology Department, Univeristy of Illinois

In an analogy a person's knowledge about one domain is used to understand a second domain: to highlight important similarities between domains or to predict new features of the second domain. For example, we use our knowledge about water flow to elucidate properties of electric circuitry. A central fact about analogy is that it is selective. Not all commonalities are equally important in matching or prediction. It follows that a central problem in theories of analogical mapping is to identify the selection constraints that tell us which similarities and dissimilarities count in an analogy (Kedar-Cabelli, 1985a; Hall, 1988). Within the enormous space of possible similarities between domains, what constrains the choice of information to include in an analogy?

Two major classes of selection constraints have been utilized in computational models of analogy: goal-relevance constraints and structural constraints. The goal-relevance constraints direct analogical mapping toward information that is relevant to the problem at hand (Burstein, 1983; Holyoak, 1985; Kedar-Cabelli, 1985b). The structural constraints focus mapping on such properties of the matching information as structural consistency (Winston, 1980; Gentner, 1980, 1983; Falkenhainer, Forbus and Gentner, in press). For example, in her account of analogy Gentner proposes that people observe the structural constraints of 1 to 1 object correspondences and structural consistency. Further, she proposes that two implicit selection rules underlie the interpretation of analogy. First, people seek to map relations between objects rather than non-relational object attributes. Second, people use a systematicity principle: they seek to map relations that participate in a system of relations governed by higher-order relations - e.g. causal relations - that can also be mapped between the two domains.

In support of the systematicity constraint, Gentner points out that the most systematic interpretation of an analogy is generally the preferred interpretation. However, Holyoak (1985) notes that systematicity is typically correlated with goal relevance and therefore these observations do not allow us to conclude that systematicity by itself can act as a selection constraint. For example, if a person who is trying to solve a problem about heat flow invokes an analogy to water flow, the causal structure relating inequality in level to direction of flow is likely to be both the most relevant and the most systematic structure shared by the two analogs. In the present research we looked at analogical mapping outside of a problem-solving context in order to test whether systematicity can act as a selection constraint on mapping in the absence of goal relevance.

We describe two experiments that focus on different aspects of analogical mapping: (a) *matching* existing information in the base and target domains (seeing similarities or identities between aspects of the base and target domains); and (b) *carryover* of new information from the base to the target (making new inferences about the target domain). Since these are distinct processes in constructing an analogy it is important to determine whether both are governed by the same selection principles. Thus the two studies tested whether systematicity acts as a selection filter to constrain both which matches are made and which inferences are drawn.

For this research we created a set of novel analogies. The basic plan was to devise analogies in which subjects could choose which of two lower-order facts to map. Both facts matched the between the base and target equally well. However they differed in whether they were part of a shared systematic relational structure. If systematicity acts as a selection constraint then subjects should focus on the fact governed by the shared structure in their interpretations and predictions from the analogy. In developing these materials we were guided by two further criteria:

1. Goal-relevance should not be a possible selection constraint. Thus, subjects did not have to map information in order to solve problems or prove points in the target domain.
2. Subjects' prior factual knowledge about the domains or expectations about relevance

should not influence their mapping. Thus, we developed analogies between novel, fictional domains rather than using real world analogies. To preserve a realistic degree of complexity, the situations described were fairly rich in information.

Design of the materials

The analogies developed each consisted of a base and target passage describing novel objects or organisms on fictional planets. The passages were about one page long and were written in the style of an encyclopedia article. Each passage included two key paragraphs. Before describing the materials further, we give an abbreviated example of one base domain which described Tams, creatures who live on a distant planet. Key paragraph 1: The Tams live on rock and can grind and consume minerals from the rock through the constant action of their underbelly. However, periodically they run out of minerals in one spot on the rock, and therefore must stop using their underbellies while they relocate. Key paragraph 2: Although at birth the Tams have rather inefficient underbellies, eventually the underbellies adapt and develop a texture that is specially suited to the rock the Tam lives on. As a consequence, the grown Tam cannot function on a new rock.

Table 1 illustrates the relational structure of this base domain "The Tams" and an analogous target domain called "The Robots", which describes robots who use probes to gather data from planets. All subjects received the same base domain. However, as shown in Table 1, there are two versions of the target domain, each given to half the subjects for counterbalancing. The base passage contains two different causal structures (the two key paragraphs discussed above). Each structure consists of a *key fact* (shown in italics in the table) and a causal system governing that key fact. The two key facts in this base domain are (1) that the Tams sometimes stop using their underbellies, and (2) that their underbellies cannot function on new rocks . Each target domain also contains two causal structures, and in both versions of the target, the key facts in each structure match the key facts in the base: (1) the robots sometimes stop using their probes and (2) the probes cannot function on new planets.

Thus, at the level of the individual key facts, both versions of the target match the base domain perfectly. However, the causal system governing a key fact does not always match the base domain. That is, within each version of the target domain, both key facts match the base, but only one key fact is linked to a causal system that also matches the base domain. We will call this the *identically -governed* key fact. We predicted that subjects should focus on this fact in mapping.

Table 1. Relational Structure of the Base Domain "The Tams" and the Target Domain "The Robots"

Base - "The Tams"	Target - "The Robots"	
	Version 1	Version 2
Consume minerals with underbellies	Gather data with probes	Gather data with probes
Exhaust minerals in one spot and must relocate on the rock	**Exhaust data in one place and must relocate on the planet**	Internal computers over-heat when gather a lot of data
So stops using underbelly	*So stops using probes*	*So stops using probes*
Born with inefficient underbelly	Designed with delicate probes	Designed with inefficient probes
Underbelly adapts and becomes specialized for one rock	Robots cannot pack probes to survive flight to a new planet.	**Probes adapt and become specialized for one planet**
So underbelly can't function on new rock	*So probes can't function on new planet*	*So probes can't function on new planet*

Note. Key facts are shown in italics. Matching causal information is shown in bold face. In Experiment 2 italicized facts were removed from the target.

The materials were designed to avoid confoundings with particular content. It can be seen that in Target version 1, the cause for the first key fact matches the base domain, but the cause for the second key fact does not. Target version 2 simply reverses which key fact is identically-governed. This ensures that preferences for identically-governed key facts cannot be attributed to preferences for a particular key fact in itself. A further control task was included to ensure that subjects' responses cannot be attributed to preference for a particular fact in context. That is, a key fact might seem more or less salient depending on its context in a particular version of the target, independently of the analogy. To check for this possibility, control subjects, who read only the Target domains, judged which key fact was more important for each domain. If these control subjects show no bias towards one key fact or the other, then any preference among experimental subjects can be attributed to the specific effects of the match with the base system.

Experiment 1 examined the matches people include in an analogy. Subjects had to judge how well each key fact in the target contributed to the analogy with the base domain. We predicted that identically-governed key facts would be preferred over differently-governed key facts, even though in each target both key facts match the base equally well.

Experiment 1

Method

Subjects.

Subjects were 48 paid undergraduate students at the University of Illinois. Half were assigned to the experimental group and half to the control group. Subjects were run in groups of three to six people.

Task and Hypotheses.

Experimental subjects. The subjects' task was to study the base and target domains and then judge how well each key fact in the target contributed to the analogy. Since our materials are complex, subjects were given three *learning* tasks before judging the key facts to ensure that they processed the analogy thoroughly: (a) they read the base and target domains. (b) they identified which object in the target domain corresponded to each object from the base domain. (c) they described the ways in which the two domains were and were not analogous. Two judgment tasks were then given:

Rating Task. Subjects were given the two key facts from the target domain (e.g. the italicized facts from the Target in Table 1). Subjects rated on a scale of 1 to 7 how well each fact "supports the claim that the base and target were analogous; how well does each fact contribute to making the analogy a good analogy".

Choice Task. Subjects were again given the two key facts and chose which *best* supported the analogy.

During the learning and judgment tasks for each analogy subjects were encouraged to re-examine the descriptions of the two domains as needed. Thus, there were no memory requirements. We predicted that identically-governed key facts should receive higher ratings than differently-governed key facts, and should be more often chosen as the fact that best supports the analogy.

Control subjects. Control subjects read and summarized the target domain. Following this, subjects were given the two key facts and were asked to (a) rate on a scale of 1 to 7 how important each fact was to the target domain and (b) choose which of the key facts was most important. Assuming the materials are not biased, the control subjects should show no preference for either identically or differently-governed key facts.

Materials.

Subjects completed these tasks for four novel analogies each designed according to the structure described above. Analogies were given in four orders of presentation across subjects. For each analogy, all subjects received the same base version; there were two target versions which were each given to half the subjects in each group (see Table 1).

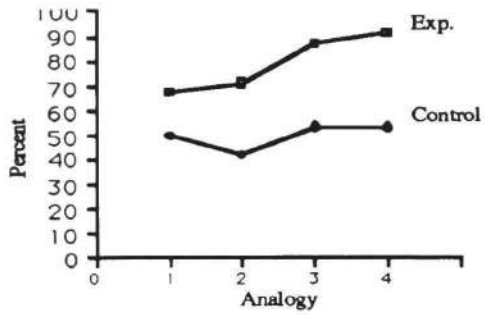


Figure 1. Exp.1 Choice Task: The Percentage of Subjects in Each Group Choosing the Identically Governed Key Fact for Each Analogy.

Results and Discussion

Rating Task

As predicted, experimental subjects rated identically-governed key facts significantly higher than differently-governed key facts. The mean ratings across the four analogies were 6.28 and 4.85 for each fact type respectively. The control subjects gave equivalent importance ratings to the two sets of key facts (M = 5.2 and 5.3) indicating that the experimental results are not due to differences in the importance of the key facts in the two versions of the target.[1]

Choice Task

As shown in Figure 1 the results again support our prediction that the subjects should view the identically governed key fact as supporting the analogy. Across the four analogies, the experimental subjects chose the identically-governed fact 67% to 92% of the time, while control subjects chose it 42% to 54% of the time (i.e. at chance level). [2]

The results of Experiment 1 support the claim that systematicity acts as a selection constraint on which matches are selected as belonging to an analogy. Even though both key target facts matched the base domain equally well, the one that was linked to a larger causal structure that also matched the base domain was consistently judged to contribute best to the analogy. Further, the lack of preference in the control group indicates that the results are not due to the importance of a fact in the target domain.

Experiment 2

Experiment 1 focused on which of two matching facts would be selected as belonging to an analogy. The second experiment examined the carryover process in analogical mapping. We deleted certain key facts from the target passages in order to test whether systematicity would act as a selection constraint on subjects' predictions about a target domain.

Method

Subjects

Subjects were 32 (16 control and 16 experimental) paid undergraduate students at the University of Illinois.

Materials and Design

The base and target domains were identical to those of Experiment 1 except that the two key facts were removed from the target domains. Thus, in Table 1, the two italicized facts shown in each target were removed. This meant that the target domain included only the antecedent information that could potentially cause each key fact in that domain. As in Experiment 1, two versions of each target were used. In each version, one antecedent matched the base domain but the other antecedent did not.

Subjects were asked to make a prediction about the target domain. The question was whether they would predict one of the two key facts omitted from the target, and more importantly, which one they would predict. If subjects are simply trying to predict facts about the target that correspond to facts in the base, they should be equally likely to predict either of the omitted key facts. However, the systematicity prediction is that subjects should predict the one key fact for which there was a matching causal system, ie. identically-governed in the base and target.

Task

Experimental subjects. For each analogy subjects first had three learning tasks identical to those in Experiment 1. They were then given three experimental tasks:

Prediction Task. Subjects were asked to make one prediction about the Target domain. They were told to predict new information about the target "that was suggested by the analogy with the base domain".

Rating and Choice Tasks. After making their own predictions subjects were asked to judge possible predictions about the target. Subjects were given as predictions the two key facts that appeared in the base but not in the target. They rated each of these on a scale of 1 to 7 according to how well this prediction followed from the analogy. Finally, subjects were again given the two key facts and asked to choose which was best predicted by the analogy.

Control subjects. Control subjects made predictions after reading only the target domain. These subjects read and summarized the target and then performed Prediction, Rating and Choice tasks parallel to those given the experimental subjects. Control subjects predicted one new fact about the target "that might be true given the information about the target". Then, given the two key facts as predictions about the target, subjects rated each prediction "according to how well it followed from information in the target" and chose which prediction best followed from the information in the target.

Results and Discussion

Prediction Task

As predicted, experimental subjects tended to make the prediction sanctioned by systematicity. Figure 2 shows subjects' predictions about the target domain grouped into three categories: (a) predictions of key facts that would be identically-governed in the base and target (b) predictions of key facts that would be differently-governed (c) all other predictions about the target. As shown, experimental subjects most frequently predicted the identically-governed key fact. In contrast, the most frequent response for the control subjects was to predict other information, and their remaining responses were evenly distributed between the two key facts. The difference in performance between the control and experimental subjects indicates that the experimental results are not due to a bias in which one of the key facts was inherently more plausible in the target. More fundamentally, it indicates that the experimental subjects were indeed

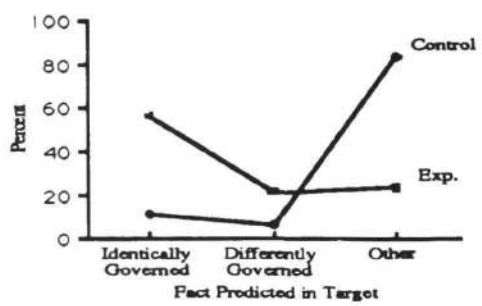


Figure 2. Exp. 2 Prediction Task: Percent of Predictions in Each Category in Each Group

guided by the analogy with the base. [3]

Rating and Choice Tasks

As hypothesized, experimental subjects rated identically-governed predictions significantly higher than differently-governed predictions (M = 6.03 and M = 4.41, respectively). Control subjects showed the reverse pattern, again indicating that materials were not biased in favor of the hypothesis (M = 4.32 and M = 5.22 respectively).[4] The results for the choice task also support the systematicity hypotheses. The experimental subjects chose the identically governed fact 62% to 87% of the time, while control subjects chose it 25% to 66% of the time (they predicted at or below chance level -50%- for three of the four analogies). [5] Thus, although for each analogy both potential predictions corresponded to facts given in the base domain, the one prediction linked to a matching causal system was judged by subjects to follow best from the analogy.

General Discussion

The results show that systematicity is a selection constraint on the choice of information to map in an analogy. In both experiments the subjects were provided with two possible matching facts or predictions, both of which matched well between the base and target, but only one of which participated in a governing higher-order system that also matched across domains. In Experiment 1 subjects chose which of these key matches contributed best to the analogy; they consistently preferred the fact linked to the larger matching system. In Experiment 2 subjects made predictions about the target domain. Their predictions, like their judgments of matches centered on the particular key facts that would follow from a matching causal antecedent.

Although the issue of selection constraints on analogy has attracted attention in AI (Winston, 1980; Carbonell, 1981), psychological research on this topic is just beginning. There is prior evidence that systematicity affects the subjective evaluation of the soundness of an analogy (Gentner & Landers, 1985; Rattermann & Gentner, 1987). However, to our knowledge the present research is the first test of the stronger claim that systematicity can act as a selection filter on mapping. This study provides evidence that systematicity can operate as a selection constraint in tasks in which no prior goal context exists, i.e. in which there is no problem to be solved so that relevance to existing goals cannot determine mapping choices.

It is important to note, however, that these results do not address whether goal-relevance can act as a filter instead of, or in addition to, structural constraints. Burstein and Adelson (1987) give protocol evidence that goal relevance constrained the selection of systematic information to map in an analogy used to solve computer programming problems. Thus goal relevance may be used to disambiguate analogous cases where the structural information is not sufficient, much as knowledge of the surrounding context is used to disambiguate sentence meanings in cases where syntactic information is insufficient, as in "flying planes can be dangerous".

Relevance may interact with structural constraints in several ways. For example, before analogical mapping begins task goals may govern the content of the domain representations that are inputs to the analogy process (Gentner, 1988). Or, as discussed above if multiple matching systematic structures are found, the choice among them may be based on relevance to current goals. These two constraints have been integrated to some extent in some computational models of analogy (Burstein, 1983; Kedar Cabelli, 1985). In any case our results reinforce the need to distinguish the structural coherence of information to be mapped from the relevance of this information to a current task goal. Although contextual relevance will certainly play a role in the interpretation of an analogy, the systematicity of shared information is itself a psychologically real constraint on mapping.

Footnotes:

1. A repeated measures ANOVA comparing the mean ratings of identically governed and differently governed key facts by each group revealed a significant interaction between group and fact -type F' (1,34)=14.37, $p < .001$. The analyses of the simple effects indicates that the difference between ratings of each fact type within the experimental group is significant, F' (1,9)=10.86, $p < .01$

2. The difference between groups is significant for three of the four analogies: ($p < .05$) Fisher Exact tests, 1 tailed. Another scoring method is to assign each subject a score for the number of choices of identically-governed key facts across the four analogies (possible score is 0-4), the mean for experimental subjects (3.17) is significantly greater than the mean for control subjects (2.0), $F(1,28) = 18.67$, $p < .0001$.

3. Agreement between these ratings and those of a second, blind judge, was .91. A repeated measures anova, comparing the mean number of identically governed and differently governed predictions by each group shows a significant interaction between group and fact type ($F(1,16) = 11.76$, $p < .01$). The analyses of the simple effects indicates that the difference between predictions of each fact type within the experimental group is significant $F(1,8) = 36$, $p<.001$. The difference within the control group is not significant.

4. The interaction between group and fact-type is significant $F'(1,11) = 9.0$, $p < .025$. The analyses of the simple effects indicates that the difference between ratings of each fact type within the experimental group is significant, $F'(1,11) = 6.05$, $p < .05$.

5. When subjects are scored for the number of choices of identically-governed key facts across the four analogies (possible score is 0-4), the mean for experimental subjects (2.9) is significantly different from the mean for control subjects (1.8), $F(1,16) = 16.2$, $p=.001$

References

Burstein, M. (1983). Concept formation by incremental analogical reasoning and debugging. In R. Michalski, J. Carbonell, & T. Mitchell (Eds.), Machine Learning (Vol. 2, pp. 351-369). Los Altos, CA: Morgan Kaufman.

Burstein, M., & Adelson, B. (1987). Analogical learning: Mapping and integrating partial mental models. In Program of the Ninth Annual Conference of the Cognitive Science Society (pp. 11-22). Hillsdale, NJ: Lawrence Erlbaum Associates.

Carbonell, J. G. (1981), Invariance hierarchies in metaphor interpretation, Proceedings of the Third Annual Meeting of the Cognitive Science Society, Berkeley, CA, pp. 292-295.

Falkenhainer, B., Forbus, K. D. & Gentner, D. (in press). The structure-mapping engine Artificial Intelligence.

Gentner, D. (1980). The structure of analogical models in science (Tech. Rep. No. 4451). Cambridge, MA: Bolt Beranek and Newman, Inc.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. Cognitive Science, 7 (2), pp. 155-170.

Gentner, D. (1988). Mechanisms of analogical learning. To appear in S. Vosniadou and A. Ortony, (Eds.), Similarity and analogical reasoning.

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, (pp.607-613). Tucson, AZ.

Hall, R. P. (1988). Computational approaches to analogical reasoning: A comparative analysis. Artificial Intelligence, pp. 1-75.

Holyoak, K. J. (1985). The pragmatics of analogical transfer. In G. H. Bower (Ed.), The Psychology of Learning and Motivation, (Vol. 1, pp. 59-87). New York: Academic Press.

Kedar-Cabelli, S. (1985a). Analogy - from a unified perspective. (Tech. Rep. No. ML-TR-3). New Brunswick, New Jersey. Laboratory for Computer Science Research, Hill Center for the Mathematical Sciences, Rutgers University.

Kedar-Cabelli, S. (1985b). Purpose-directed analogy. Proceedings of the Seventh Annual Conference of the Cognitive Science Society (pp. 150-159), Irvine California.

Rattermann, M. J. & Gentner, D. (1987). Analogy and similarity: Determinants of accessibility and inferential soundness. In the Program of the Ninth Annual Conference of the Cognitive Science Society pp. 23-34.

Winston, P. H. (1980). Learning and reasoning by analogy. Communications of the ACM, 23 (12), pp. 689-703.

Acknowledgements

This research was supported by a Child Health and Human Development Grant No. HD07205-08, and by the Office of Naval Research under Contract No. N00014-85-K-0559. We thank Laura Kotovsky, Ronald Mawby, Betsy Perry, Mary Jo Rattermann, Brian Ross, and Bob Schumacher.

ABSTRACTION PROCESSES DURING CONCEPT LEARNING: A STRUCTURAL VIEW

Janice Skorstad
Computer Science Department
University of Illinois

Dedre Gentner
Psychology Department
University of Illinois

Doug Medin
Psychology Department
University of Illinois

INTRODUCTION

Learning concepts from examples is a central process in cognition. In the psychological literature, this process is known as *schema-abstraction*. We focus here on the phenomenon of **sequence effects**, ie., the effect presentation order has upon the concept learned. This paper describes a computer model, **SEQL,** which provides an experimental tool for exploring a class of schema-abstraction theories. We describe the organization of SEQL, illustrating how structural comparisons combined with a library of abstraction strategies can model sequence learning effects. We briefly describe a psychological experiment designed to explore sequence effects and show how SEQL can be used to reproduce the results.

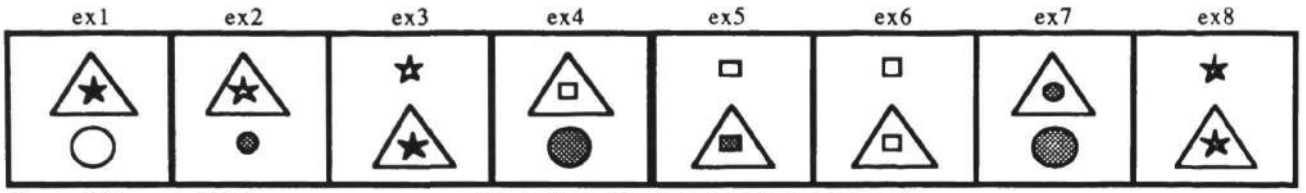
Schema-abstraction theories assume that some information is abstracted during learning and is subsequently stored for later use during classification of new examples. These theories can be differentiated on the basis of three criteria: how the abstracted concept information is (i) **characterized**, (ii) **retained** and (iii) **utilized** to classify new instances. For example, the *prototype* theory [Posner & Keele 68, Posner & Keele 70] assumes that during learning, subjects construct a single representation of the concept's prototype by calculating the average of all the training instances. New exemplars are classified by determining how similar they are to the prototype. Posner & Keele found that classification of never-studied prototypes was more accurate than classification of never-studied exemplars. In fact, after delay there was a greater loss in classification accuracy for the much-studied exemplars than the never-studied prototypes.

In contrast, Medin & Schaffer [Medin & Schaffer 78] proposed an *instance-only* model of concept-learning. This model posits that no abstraction is performed, and only training exemplars are stored. Classification of new training instances is based on their similarity to all of the stored items. Medin & Schaffer demonstrated that the effects arising from schema-abstraction models—namely, the superior performance of prototypes and the increase in this effect after delay, can be accounted for by their instance-only model. In view of this, it has proved difficult to predict differences in the performance of schema-abstraction versus instance-only models.

One phenomenon that might allow us to differentiate these models (and also subclasses of schema-abstraction models) is sequence effects. Such effects are impossible for a standard instance-only model since transfer stimuli are compared to <u>all</u> of the items in memory. These effects therefore might provide evidence for a schema-abstraction model. Unfortunately, it is difficult to predict the performance of schema-abstraction models in concept learning since we lack an explicit definition of the abstraction process.

We aim to attack the problem on two fronts, psychological and computational. We have designed a human experiment and a computer simulation to examine possible abstraction processes.

Group I Sequence:



Group I Expected Rule:

(IF (AND (OUTSIDE X TRIANGLE)
(EQUALS (SHAPE X) CIRCLE))
(EQUALS (LOCATION TRIANGLE) TOP))

(If the figure outside of triangle is a circle, then the triangle is on top.)

Figure 1.

Kline [Kline 83] demonstrated that human learning is indeed affected by presentation sequence. So far, our attempt to replicate Kline has produced weak results. In spite of this, we describe our experiment in order to present the logic behind our reasoning and to show the materials we used.

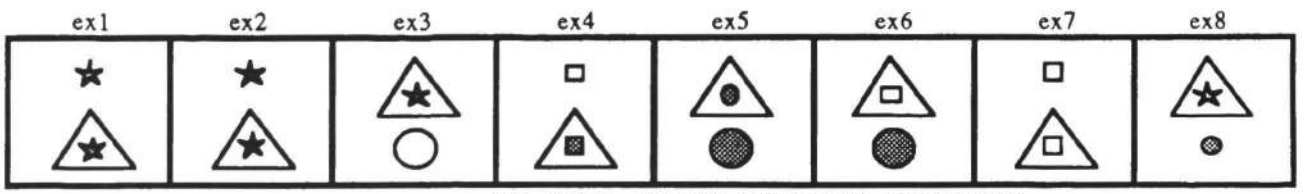
Psychological experiment

This experiment is the first in a series designed to determine if the order of training instances affects the learner's concept description. Subjects first took part in a study phase followed by a transfer phase. Subjects were divided into two groups. We diverged from the standard experimental paradigm since both groups did not simultaneously see the same training instances. Using a method similar to Kline's, each group saw the <u>same</u> training instances but the <u>order</u> of these instances varied between the two groups. Figures 1 & 2 show the two sequences of training instances along with the concept descriptions we expected subjects to generate. Unlike Kline's experiments, subjects did not have full memory; once a new exemplar was displayed, they could not go back and look at previous ones.

Experiment design

Study Phase: Subjects were first told that they would be asked to generate a rule for the figures they were about to see. They were shown 8 figures, one at a time. They were then asked to describe the rule for the figures they just saw. This was repeated four times.

Group II Sequence:



Group II Expected Rule:

(IF (AND (SAME (SHAPE X) (SHAPE Y))
(SAME (SIZE X) (SIZE Y)))
(EQUALS (LOCATION TRIANGLE) BOTTOM))

(If the figures inside and outside of triangle are the same shape and size, then triangle is on bottom.)

Figure 2.

Transfer Phase: Subjects saw 27 pairs of stimuli. For each pair, they were asked to mark which of the two was more similar to the items they saw in the study phase. Subjects saw the same 27 pairs again. (Subjects were told that these were the same 27 pairs they had just seen.) This time, in addition to marking which was more similar to study items, subjects were asked to explain their choice.

Results

The results were not strong, nor were they uniform within the groups. The difference between the groups was not significant. However, group 2 performed as expected; 55% of group 2 subjects chose rule 2. Group 1 subjects showed no preference for rule 1 over rule 2. An ANOVA did reveal a significant difference between the two rules (**). Rule 2 was found to be easier to learn than rule 1. In light of these results, which are somewhat promising, we modified the training instances so that rule 1 would be as easily learned as rule 2. [Skorstad 88] will provide a more detailed description of the experiment and its follow-up.

SEQL computer model

SEQL is being built as a tool for exploring various abstraction theories. Its results can be compared with human performance in the learning experiment. Our approach differs from many prior schema-abstraction theories. We assume that the prototype must be based on structural similarity comparisons rather than feature-set intersection. We use a model of similarity as defined by structure-mapping theory [Gentner 83]. For this purpose we use SME [Falkenhainer, Forbus & Gentner 86], a computer implementation of the structure-mapping theory.

System Description

SEQL consists of several main modules:

— **SME** operates on two potential analogs, the *base* and the *target*, generating a number of plausible mappings (*gmaps*) for these analogs. These gmaps correspond to interpretations of the analogy. In addition, SME produces an evaluation score for each of the gmaps. An important feature of SME is its toolkit approach. By using different sets of match rules, we can perform the various similarity comparisons defined by structure-mapping theory. In this research, we use the Literal-Similarity rules. By literal-similarity, we mean a similarity match in which both the objects and the relational structure of these objects are counted in the match.

— **SE** (Structural Evaluator) is a post-SME process. It provides alternate gmap evaluation scores, separate from SME's evaluator. For example, one can choose to evaluate a gmap based on a combination of its depth and breadth or on its depth and breadth <u>relative</u> to the depth and breadth of its base.

— **Generalize** finds the most specific conjunctive generalization that characterizes both the base and target descriptions fed to SME. Generalize takes a single gmap as input and outputs a generalized description of this gmap both in human readable form and in SME's syntax.

— **Specialize** modifies a generalization when it is no longer adequate to describe a new example. This occurs when the generalization is poorly matched to a target example. If the generalization's rule description (the human readable form) is conjunctive, then the generalization must be specialized. It needs to be modified in such a way so that it covers a smaller subset of examples. This can be done by transforming the conjunctive rule description into an IF-THEN rule roughly in the style proposed by Bettger [Bettger, in

preparation]. If the generalization rule was already in IF-THEN form, no specialization is necessary.

Library of abstraction strategies to be simulated

In this section, we lay out a space of possible abstraction strategies. This library of strategies will give us the flexibility to explore various abstraction theories. These strategies determine what is generalized, and when this generalization takes place. We divide the abstraction strategies into two classes, the **limiting cases** versus the **combination models**. First we list a set of limiting cases which are useful in delimiting the space of basic abstraction theories. These models are probably too simple to be used for direct comparison to human performance during the study phase. Their effectiveness or power becomes more apparent during the transfer task when the stored knowledge is retrieved and utilized to classify new instances. Next, we list the combination models. These use a parameter, **T**, the threshold value. If the evaluation score of a match is below **T**, the match is not considered to be a good one. In this way, the user has control over what he considers to be a good match.

A. Limiting Case Models
   1. Exemplar Only
      Store exemplars only. No generalizations are made during learning. This is the simplest form of the instance-only model.
   2. Radical Generalization
      Continuously generalize each new exemplar with the base (which is a generalization) to generate one single, "winning" generalization. Some versions of the prototype models would fall under this category.
   3. All Exemplars & All Generalizations
      Store all exemplars and all generalizations (which are formed for each match). This puts a heavy burden on memory. If there are n items in the study sequence, $(2^n - 1)$ items will be stored.

B. Combination Models
   1. Generalization and Exemplar
      If the match is greater than **T**, store their generalization, the base, and the target, otherwise store the base and target only (up to a memory limit).
   2. Generalization or Exemplar
      If the match between the base and target is greater than **T**, store their generalization, otherwise store the base and target (up to a memory limit).
   3. IF-THEN-Rule-Generator
      This is a specialization of model B.2. In addition to storing the base and target when a bad match is found, the generalization rule is modified to form an IF-THEN rule. The psychological motivation for this strategy is the intuitive notion that when humans encounter an exception during concept learning, ie., when an example is found which isn't covered by their rule, they focus on the differences in the base and target. They use these features to patch up their current rule description [Medin personal communication 87 & Bettger in preparation].

One of the goals of our research is to test each of these abstraction strategies on the same set of examples. We illustrate the **IF-THEN-Rule-Generator** strategy in more detail in the next section.

Ex1	Ex2		Gen[1,2]
(star obj1)	(star obj1)		(star entity17)
(star obj2)	(triangle obj2)		(star entity24)
(triangle obj3)	(star obj3)		(triangle entity33)
(small obj1)	(small obj1)		(small entity17)
(small obj2)	(large obj2)		(small entity24)
(large obj3)	(small obj3)	→	(large entity33)
(above obj1 obj2)	(above obj1 obj2)		(above entity17 entity24)
(above obj1 obj3)	(above obj1 obj3)		(above entity17 entity33)
(inside obj2 obj3)	(inside obj3 obj2)		(inside entity24 entity33)
(outside obj1 obj3)	(outside obj1 obj2)		(outside entity17 entity33)
(bottom obj3)	(bottom obj2)		(bottom entity33)
(same-shape obj1 obj2)	(same-shape obj1 obj3)		(same-shape entity17 entity24)
(same-size obj1 obj2)	(same-size obj1 obj3)		(same-size entity17 entity24)

Figure 3.

SEQL example using IF-THEN-Rule-Generator strategy

Suppose our goal is to generate a concept description for the sequence of 8 exemplars shown in figure 2. The representations for exemplar 1 and 2 are shown in figure 3. (For compactness, we removed some of the attributes). **Ex1** and **ex2** are input to SME. **Ex1** is the base and **ex2** the target. SME generates a number of plausible mappings (gmaps) between these two exemplars. Running the Structural Evaluator on these mappings yields a single best normalized gmap which is compared with the threshold value to see if it is a good match. For these two exemplars, we have a good match. The gmap is passed on to the generalizer which produces the generalization, **gen[1,2]** shown in figure 3.

The generalization, **gen[1,2]** and the new exemplar, **ex3**, (see fig 4) become the next base and target respectively. They don't yield a good match. The highest scoring normalized gmap is less than the threshold value. **Gen[1,2]**'s rule description must be specialized since it no longer describes all of

Gen[1,2]	Ex3		IF-THEN Rule (gen [1,2], ex3)
(star entity17)	(star obj1)	(IF (AND	(star entity24)
(star entity24)	(circle obj2)		(small entity24)
(triangle entity33)	(triangle obj3)		(above entity17 entity24)
(small entity17)	(small obj1)		(inside entity24 entity33)
(small entity24)	(medium obj2)		(same-shape entity17 entity24)
(large entity33)	(large obj3)	→	(same-size entity17 entity24))
(above entity17 entity24)	(bottom obj3)	(THEN	
(above entity17 entity33)	(above obj1 obj2)	(AND	(triangle entity33)
(inside entity24 entity33)	(above obj3 obj2)		(large entity33)
(outside entity17 entity33)	(inside obj1 obj3)		(star entity17)
(bottom entity33)	(outside obj2 obj3)		(small entity17)
(same-shape entity17 entity24)			(above entity17 entity33)
(same-size entity17 entity24)			(outside entity17 entity33)
			(bottom entity33))))

Figure 4.

processes that occur during concept learning.

Finally, we have shown how SEQL can simulate the phenomenon of sequence effects.

Acknowledgements
Ken Forbus & Gordon Skorstad provided valuable suggestions, both in the research and in the writing of this paper. Brian Falkenhainer provided many useful comments and suggestions on the research. This research is supported by the Office of Naval Research, Contract No. N00014-85-K-0559.

References

Bettger, J. (in preparation), Master's Thesis, University of Illinois, Department of Psychology.

Elio, R., & J.R. Anderson (1981), "The Effects of Category Generalizations and Instance Similarity on Schema Abstraction", *Journal of Experimental Psychology: Human Learning and Memory*, **7** (6), 397-417.

Elio, R., & J.R. Anderson (1984), "The Effects of Information Order and Learning Mode on Schema Abstraction", *Memory & Cognition*, **12** (1), 20-30.

Falkenhainer, B., K.D. Forbus, D. Gentner (1986), "The Structure-Mapping Engine", *Proceedings of the Fifth National Conference on Artificial Intelligence*, August.

Gentner, D. (1983), "Structure-Mapping: A Theoretical Framework for Analogy", *Cognitive Science* **7**(2).

Hayes-Roth, F., & J. McDermott (1978), "An Interference Matching Technique for Inducing Abstractions", *Communications of the ACM*, **21**, 5, 401-411

Kline, P.J. (1983), "Computing the Similarity of Structured Objects by Means of a Heuristic Search for Correspondences", Doctoral Dissertation, University of Michigan, Department of Psychology.

Medin, D.L., & M.M. Schaffer (1978), "Context Theory of Classification Learning", *Psychological Review*, **85** (3), 207-238.

Posner, M.I., & Keele, S.W. (1968), "On the genesis of abstract ideas", *Journal of Experimental Psychology*, **77**, 353-863.

Posner, M.I., & Keele, S.W. (1970), "Retention of Abstract Ideas", *Journal of Experimental Psychology*, **83**, 304-308.

Ross, B.H., S.J. Perkins & P.L. Tenpenny, (submitted for publication, 1987), "Reminding-Based Category Learning".

Skorstad, J.C. (in preparation), Master's Thesis, to appear as a Technical Report, University of Illinois, Department of Computer Science.

```
(IF (AND (small entity24)
         (inside entity24 entity33)
         (same-shape entity17 entity24)
         (same-size entity17 entity24))
    (THEN
     (AND (triangle entity33)
          (large entity33)
          (above entity17 entity24)
          (small entity17)
          (above entity17 entity33)
          (outside entity17 entity33)
          (bottom entity33))))
```

IF-THEN Rule (sequence 2)

```
(IF (AND (circle entity64)
         (above entity12 entity64)
         (above entity70 entity64)
         (outside entity64 entity12))
    (THEN
     (AND (triangle entity12)
          (large entity12)
          (top entity12)
          (small entity70)
          (inside entity70 entity12)))
```

IF-THEN Rule (sequence 1)

Figure 5.

the exemplars seen. Specializing **gen[1,2]** produces the new rule description, **IF-THEN Rule (gen[1,2], ex3)** shown in figure 4. This description asserts that if there is a small star inside another object, with an object of the same shape and size above it, then there is large triangle located at the bottom of the frame, with a small star above and outside it. This human-readable IF-THEN rule is kept in addition to the original **gen[1,2]** assertions. In the next match, it is these assertions, **gen[1,2],** that are input to SME—not the IF-THEN rule. Running the remainder of the exemplars through SEQL and displaying the generalization with the highest weight, we get the concept description, **IF-THEN Rule (sequence 2).** (see figure 5)

IF-THEN Rule (sequence 2) asserts that if there is a small object inside another object, with an object of the same shape and size above it, then there is large triangle located at the bottom portion of the frame, with a small object above and outside it. This description corresponds to the description that 55% of the subjects from group 2 of our experiment generated.

When the <u>same</u> set of exemplars are presented to SEQL in a <u>different</u> order, a different rule description is generated, thus demonstrating a sequence effect. For example, if the sequence shown in figure 1 is run through SEQL, the rule **IF-THEN Rule (sequence 1)** listed in figure 5 is produced. This rule states that if there is a circle which is below two entities and outside of one of the entities, then there is a large triangle located at the top of the frame with a small object inside it.

SEQL's results are considerably "better" than those generated by our human subjects, especially those in group 1. Hopefully, the modifications we're making to our experiment will improve subjects' performance. If not, this will discount the psychological validity of our model.

Discussion

Current work in concept-formation suggests that abstraction does indeed take place during concept learning [Ross et. al. submitted for publication 87, Elio & Anderson 84, Kline 83]. Our delineation of a space of some of these possible abstraction processes enables us to capture and avoid some of the extremes of the prototype and instance-only models.

An important distinction of our approach is the use of structural comparisons defined by structure-mapping theory as opposed to the more typical feature-set intersection. We have shown how SEQL incorporates structural comparisons to produce a useful tool for exploring abstraction

EXPLANATORY COHERENCE AND BELIEF REVISION IN NAIVE PHYSICS

Michael Ranney and Paul Thagard

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

Students of reasoning have long tried to understand how people revise systems of beliefs (see Wertheimer, 1945, for example). We will describe a computational model of how experimental subjects revise their naive beliefs about physical motion. We maintain that people often change their beliefs in ways driven by considerations of explanatory coherence. After describing instances in which subjects change their beliefs while learning elementary physics, we show how their belief revisions can be modeled using ECHO, a connectionist computer program that uses constraint-satisfaction techniques to implement a theory of explanatory coherence.

THE PHENOMENA: CHANGES IN SYSTEMS OF BELIEFS

Ranney (1987a) investigated belief change in naive subjects learning elementary physics by using feedback provided on a computer display. Subjects were asked to predict the motion of several projectiles and then explain these predictions. The physical contexts were quite simple, involving objects that were either thrown or released in various ways. Analyses of verbal protocol data indicate that subjects sometimes underwent dramatic belief revisions while offering predictions or receiving empirical feedback. We will describe two kinds of revisions.

Pat's Changes

Consider "Pat," an individual who was asked to offer predictions about events including (a) the motion of a heavy object dropped by a briskly walking man and (b) the motion of a heavy object thrown *obliquely* upward. Using episodic memories and mental imagery, Pat initially predicted that the object dropped by the man would fall straight down (relative to the ground). This belief is a common finding in the naive physics literature (McCloskey, Washburn, & Felch, 1983). Although she entertained the correct prediction, that the dropped object might curve forward due to the object's forward "force" (velocity), she preferred to stay with the straight-down belief.

Several tasks later, when faced with the "upward-throw" situation, Pat noted a similarity between it and the "walking-drop" task -- one that eventually spawned a belief revision. While she offered the correct parabolic trajectory as a prediction for the upward-throw, she noted that, at the parabola's zenith, the upwardly thrown object is comparable to that just released by the walking man. That is, at the apex of the thrown object's trajectory, it has an exactly-horizontal motion, as does the just-dropped object. Pat then mentioned that this observation was not "consistent" with what she said before and, if she *were* to be consistent, the thrown object would "stop" its horizontal motion and "then just fall straight down" from the zenith of the parabola. This "curving-up-then-straight-down" trajectory was not consistent with her past experience of falling objects.

Pat then realized that her memory-driven description of the ball dropping straight down from the walking man involved beliefs that were incoherent with her beliefs about the parabolic motion of thrown bodies. After a period of ignoring the incoherence, Pat stated that she had "constructed a consistent theory of how these things move." Remarkably, she went on to reject her straight-down prediction for the walking-drop task and accept the belief that the path would have a "slight forward" arc combining the "forward force" and gravity. Eventually, Pat generalized this notion, discriminating among the breadths of the arcs of several laterally released projectiles.

Hal's Changes

A second kind of systematic belief revision occurred in subjects who offered predictions, received feedback, and provided explanations for a set of tasks in which pendulum-bobs were released from their supportive strings during various points in a swing. This set of tasks was adapted from stimuli used by Caramazza, McCloskey, & Green, (1981). Because of the similarity among several of the subjects, we will amalgamate them into a composite subject "Hal."

Hal predicted that, at the extreme "endpoint" of a swing, a released bob will travel laterally and (even-

tually) downward. To some extent, this prediction was driven by images of children flying off playground swings. Via feedback, Hal learned that a bob released in this manner actually falls straight down (see Figure 1, position E). Most of the subjects observed by Ranney (1987a) were surprised by this piece of feedback, as almost 90% of the predicted trajectories were nonvertical. Virtually all these subjects revised some beliefs, offering explanations similar to the following prototype:

> Unlike the bobs with the other release-positions, this bob went directly straight down, not to the side at all. Since it had no lateral motion as it fell, this means that the object had no speed when it was released. Therefore, the pendulum must have been temporarily stopped when the bob dropped. This makes sense, since the pendulum was probably slowing down -- and it had to stop in order to change directions!

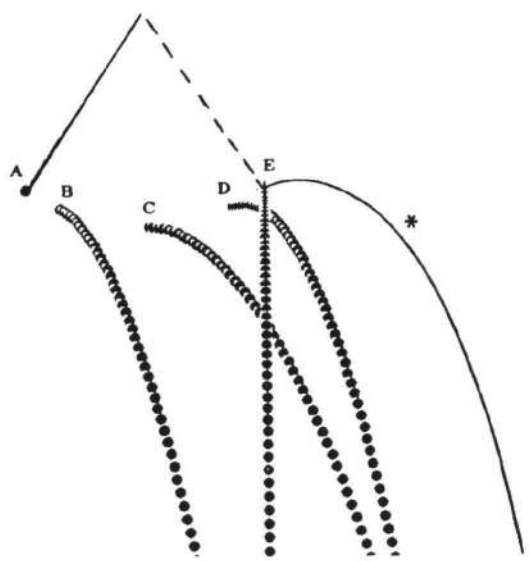


Fig. 1. Hal's prediction (*) and four feedback paths.

In contrast to Pat's belief change, in which two incoherent predictions caused her to reject one of the them, Hal's belief system underwent a more dramatic revision. He came to accept both the straight-down feedback and the notion of an instantaneous zero velocity, while rejecting both his earlier (lateral) prediction and an impetus-driven belief regarding pendular motion. (See Halloun & Hestenes, 1985, and Ranney, 1987b, for descriptions of different sorts of impetus beliefs.)

EXPLANATORY COHERENCE AS A MECHANISM FOR SYSTEMATIC BELIEF REVISION

How can we account for these systematic changes in beliefs? Both cases involved a subject's attempt to adjust beliefs in order to explain a surprising observation. An adequate model of these phenomena must provide a mechanism by which a coherent, revised, set of beliefs can arise from the need for explanation.

ECHO

Thagard (1988a) has proposed a theory of explanatory coherence that builds on previous ideas about the evaluation of explanatory hypotheses (Harman, 1986; Thagard, 1988b). The theory has been implemented in a connectionist computer program, ECHO, that uses parallel constraint satisfaction to accept and reject hypotheses on the basis of their explanatory coherence. ECHO has been used to analyze a variety of scientific arguments, past and present: Lavoisier's case for his oxygen theory against the phlogiston theory, Darwin's argument for evolution by natural selection, controversies about continental drift (Thagard & Nowak, 1988), and debates about why the dinosaurs became extinct. Application of ECHO to the belief revisions in Pat and Hal is novel in two respects. First, we are modeling subject protocols produced during experiments rather than finished arguments. Second, these models are dynamic, in that ECHO changes its coherence judgments in response to new evidence.

Space constraints permit only a sketch of the theory of explanatory coherence and its implementation (see Thagard, 1988a, for greater detail). The theory is stated using seven principles of explanatory coherence that can be summarized as follows. Principle 1, *Symmetry*, states that coherence and incoherence are symmetric relations. Principle 2, *Explanation*, says that hypotheses that together explain a piece of evidence cohere with the evidence and with each other, and that the degree of coherence decreases with the number of hypotheses used in the explanation. Principle 3, *Analogy*, attributes coherence to similar hypotheses that explain similar pieces of evidence. Principle 4, *Data Priority*, states that pieces of evidence have a degree of coherence in themselves, even though evidence can be rejected for theoretical reasons. According to principle 5, *Contradiction*, contradictory propositions are incoherent. Principles 6 and 7 claim that the explanatory coherence of a proposition or set of propositions is determined by the pairwise relations established by principles 1-5.

ECHO is a Common LISP program whose input consists of statements about the explanatory and contradictory relations among propositions. It creates units representing propositions and sets up links between pairs of propositions in accord with the above principles of explanatory coherence. If two propositions cohere because they are both arguments of a particular explanation, then ECHO sets up an excitatory link between them. If two propositions are incoherent because they contradict each other, then ECHO sets up an inhibitory link between them. In accord with the principle of data priority, propositions representing evidence receive a link from a special evidence unit. For modeling the physics students, we treat as evidence propositions based on either (a) the presence or absence of direct observations, (b) memories of such observations, or (c) facts that are well-established for the subject, such as "gravity pulls objects downward."

The mathematics underlying ECHO are straightforward. Following typical connectionist practice (Rumelhart & McClelland, 1986), each unit has an activation that is updated by considering the units that are linked to it. A unit's excitatory link with another unit whose activation is greater than 0 tends to increase the first unit's activation, while an inhibitory link with the other unit tends to decrease activation. More generally, for each unit j, the activation a_j is a continuous function of the activation of all the units linked to it, with each unit's contribution depending on the *weight* w_{ij} of the link from unit i to unit j. The activation of a unit j can be updated from time t to time t+1 using the following equation.

$$a_j(t+1) = a_j(t)(1-\theta) + \begin{cases} net_j(max - a_j(t)) & \text{if } net_j > 0 \\ net_j(a_j(t) - min) & otherwise \end{cases}$$

Here θ is a decay parameter that decrements each unit at every cycle, min is minimum activation (-1), max is maximum activation (1), and net_j is the net input to a unit. This is defined by:

$$net_j = \sum_i w_{ij} a_i(t)$$

Repeated updating cycles result in some beliefs gaining acceptance (activation > 0) while other are rejected (activation < 0). ECHO networks eventually settle into stable states in which the units have asymptotic activations that represent their coherence with other units.

Applying ECHO To Pat's Belief Revision

We have used ECHO to analyze the kinds of belief revision exhibited in the subjects described above. In each case, a contradiction among the subject's beliefs appeared to serve as the motivation for the observed changes. ECHO deals with contradictions gracefully, treating them as a pressures to change beliefs, but otherwise tolerating them. Pat's case involved a critical incoherence between two mutually exclusive predictions: a piece of evidence that was supposedly observed, and a hypothesis that was not observed, yet consistent with other observations and hypotheses.

The following is a list of Pat's initial set of propositions, as garnered from her verbal protocol of the problem-solving session. They represent her active beliefs just after she provided her straight-down prediction for the walking-drop task.

Evidence:
E1. Carried objects fall straight-down upon release.
E2. Carried objects don't fall diagonally upon release.

Negative Evidence (proposed observations that do not obtain):
NE1. Carried objects fall diagonally upon release.

Common Fact:
CF1. Gravity moves released objects downward.

Newtonian Hypotheses:
NH1. Laterally moving objects curve downward (immediately) upon release.
NH2. Released objects move forward via a forward velocity.

Alternative (non-Newtonian) Hypotheses:
AH1. Horizontally moving objects fall straight-down (immediately) upon release.
AH2. Released objects move forward via a forward "force."

The following are Pat's original verbalized explanations, manifested in ECHO as excitatory links among each of the propositions involved.

Explanations:
E1 is explained by AH1;
E2 is explained by NH1;
E2 is explained by AH1;
NE1 is explained by CF1 and AH2;
NH1 is explained by CF1 and NH2;

The next set of relations are the inconsistencies that Pat originally mentioned. Recall that the contradiction that disturbed Pat was the one between E1 and NH1; she couldn't accept both (a) that laterally-released objects curve downward and (b) that carried

objects (also being laterally-released) fall straight-down.

Contradictions:
E1 versus NH1;
E2 versus NE1;
NH1 versus AH1;

When Pat was later asked to offer a prediction for the upward-throw task, she added the following beliefs:

New Evidence:
E3. Upwardly thrown objects curve up-and-down.
E4. Upwardly thrown objects do not curve up and fall straight-down.

New Negative Evidence:
NE2. Upwardly thrown objects curve up, then fall straight-down.

Finally, Pat verbalized the following explanations and contradictions. Note that the explanation of NE2 is essentially a (higher-order) explanation of a hypothesis *by* other hypotheses:

New Explanations:
E3 is explained by NH1;
E4 is explained by NH1;
NE2 is explained by NH1 and AH1;

New Contradictions:
E3 versus AH1;
E4 versus NE2;
E4 versus AH1;

Figure 2 displays the network ECHO forms from the above explanations and contradictions, with solid lines representing symmetrical excitatory links and dashed lines representing symmetrical inhibitory links. We suggest that the figure displays the essential structural aspects of Pat's working memory during the belief change in question. The graph shows that prediction NH1 is well-supported by evidence E2, E3, and E4, as well as by fact CF1 and hypothesis NH2. Prediction E1, being a "remembered" observation, has a direct source of activation via principle (4) yet is supported only by the Aristotelian hypothesis AH1.

In order to approximate Pat's belief change, ECHO should exhibit both an initial acceptance of E1, followed by it's rejection in favor of NH1. As Figure 3 illustrates, these characteristics are indeed captured by ECHO. The activation (from -1 to +1 on the y-axis) of each node is plotted against time (from 0 to 200 cycles of activation updating). With each node initially set to zero activation, the system relaxes into more and more coherent states, such that E1's trajectory follows the desired nonmonotonic path -- rising sharply, then falling into the rejected region -- as NH1 advances and AH1 declines. The other propositions are similarly accepted or rejected (or held in limbo, as is AH2), depending upon their local coherence relationships within the overall constraint-satisfaction system. Note that the model also simulates the temporal aspect of Pat's reasoning, as the "new" propositions, E3, E4, and NE2, as well as their associated explanations and contradictions, are introduced after a brief lag (after 15 cycles). The final, most stable configuration of beliefs happens to be one that roughly corresponds to Newtonian motion. (Of course, if Pat happened to recall other evidence that supported her alternative hypotheses, this need not have been the case.)

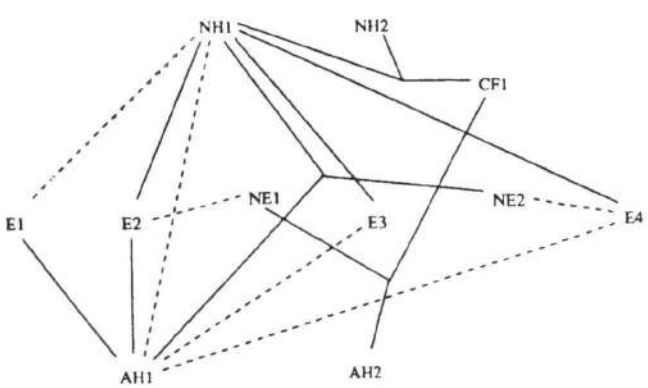


Fig. 2. Pat's explanatory coherence network.

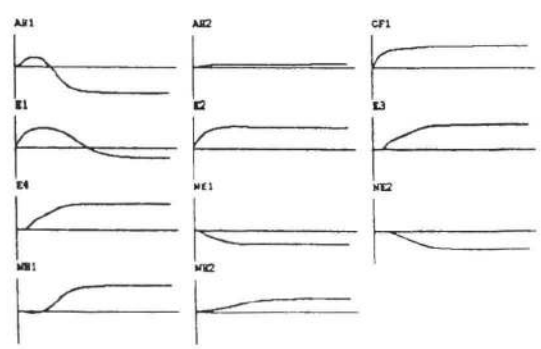


Fig. 3. The activation trajectories of Pat's beliefs.

A Dynamic Simulation Of Hal's Belief Revision

A simulation of Hal's belief changes involves a more intensive temporal analysis. Recall that Hal's revision was due to an empirically-driven contradiction, in contrast to Pat's more memory-driven contradiction. Here are Hal's essential original beliefs (i.e., his beliefs prior to receiving *any* trajectory feedback about pendulum-bobs that are released during a swing). Keep in mind that Hal is a composite subject: these are beliefs that were characteristic of many of the subjects who underwent essentially the same belief revision.

Evidence:
E1. Kids can fly off the end of a playground swing.
E5. A pendulum reverses directions at the endpoints.

Common Facts:
CF1. Gravity pulls objects downward.
CF2. A swing is a pendulum.

Classical Physical (Newtonian) Hypotheses:
CP1. At the endpoints, a pendulum is at rest.
CP2. A laterally-released object moves over and down.
CP3. The slower a pendulum-bob's speed at release, the smaller the curved trajectory.

Alternative (non-Newtonian) Hypotheses:
AH1. At the endpoints, a pendulum-bob continues its preceding lateral motion.

Predictions:
P1. At the endpoint, a released bob will move over and down.
P2. At the endpoint, a released bob will fall straight-down.

Both E1 and E5 are remembered observations. E2, E3, and E4 were intentionally left out for now, since these pieces of evidence will be sequentially added as feedback, as described later. The following explanations and contradictions were common to protocols reflecting Hal's belief revision. Note that here the critical incoherence (which feedback eventually resolves) is between P1 and P2, two mutually-exclusive predictions with different levels of support and competition.

Explanations:
E1 is explained by AH1 and CF2;
E5 is explained by CP1;
P1 is explained by AH1 and CP2;
P2 is explained by CP1 and CP3;
P2 is explained by CP1 and CF1;
CP2 is explained by CF1;

Contradictions:
E1 versus P2; P1 versus P2; CP1 versus AH1;

Figure 4 shows that, when ECHO is loaded in such a fashion at time t_0, the system reaches a stable state by t_1 (after 150 processing cycles). Among other dynamic relationships, these graphs show that P1 (the curving-down-at-endpoint prediction) is believed, while its antagonist, P2 (the straight-down-at-endpoint prediction), is disbelieved -- as indicated by its negative activation. Thus, t_1 represents the state of Hal's belief system prior to any feedback.

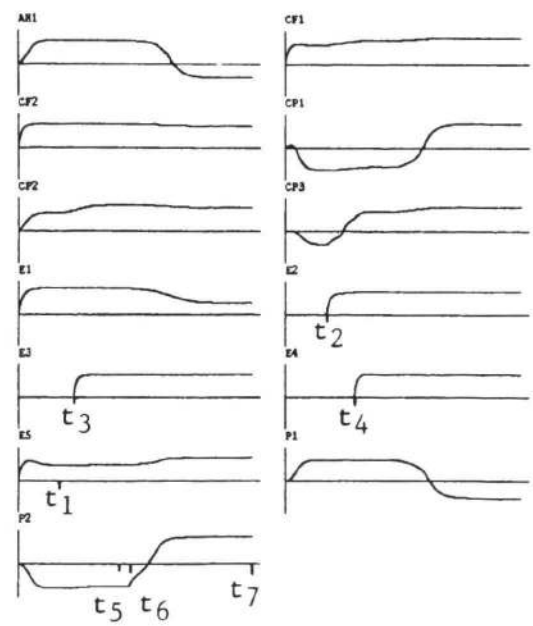


Fig. 4. The activation trajectories of Hal's beliefs.

At t_2, t_3, and t_4 (of Figure 4), evidence about other pendular-release positions is acquired in the form of direct observations (i.e., feedback) E2, E3, and E4. These "within-swing" paths are readily explained by (and hence support) propositions CF1, CP2, and CP3:

New Evidence:
E2. A bob released on a downswing curves down after its release.
E3. A bob released from midswing curves out (a lot) after its release.
E4. A bob released on an upswing curves up-and-down after its release.

New Explanations:
E2 is explained by CF1, CP2, and CP3;
E3 is explained by CF1, CP2, and CP3;
E4 is explained by CF1, CP2, and CP3;

The system then settles into state t_5 (after 400 total cycles). Figure 4 shows that, except for the generalization expressed in CP3 (relating release-velocity to the breadth of curves), little has changed from state t_1; P1 is still believed and P2 is not. Figure 5 shows Hal's belief system from t_5 onward, including all excitatory and inhibitory links.

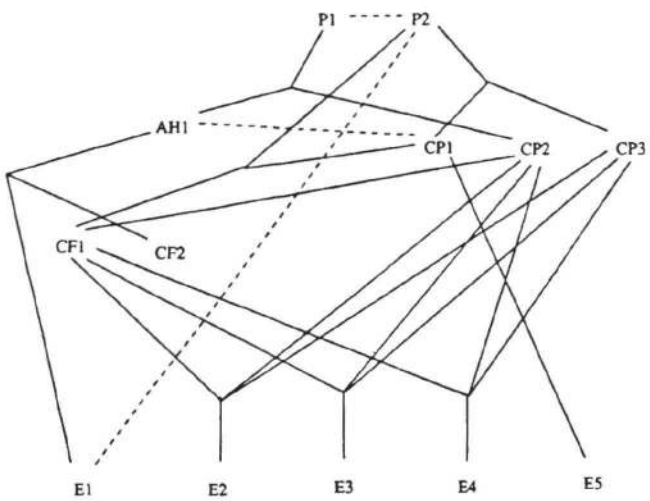


Fig. 5. Hal's explanatory coherence network.

As described earlier, it is at time t_6 that the dramatic belief revision begins, driven by the surprising feedback that, contrary to P1, the endpoint release yields a straight-down path (as predicted by the disbelieved P2). This feedback is simulated in ECHO by making P2 a data node, thus providing it with a direct source of activation (like E1-E5, CF1, and CF2 which also have data priority.) As Figure 4 indicates, this single change has five dramatic consequences between t_6 and Hal's ultimate state (after 850 total cycles), t_7: (a) P2 gains acceptance, flipping from a negative to a positive activation-state, while (b) the antagonistic P1 is rejected. (c) CP1, the notion of instantaneous zero velocity, achieves acceptance, while (d) its non-Newtonian antagonist, AH1, is rejected. (e) Even E1, a fallacious piece of "evidence" (i.e., that kids can fly off the end of swings) loses support. These changes essentially reflect the belief revisions verbalized by subjects like Hal.

ASSESSING THE MODEL

While these simulations provide general correspondence with Pat's and Hal's changes in belief, there are several methodological questions to consider. We must ask how sensitive ECHO is to (a) the particular representation of an individual's beliefs and (b) the particular parameters involved in activation-passing.

How arbitrary are the representations that are put to ECHO? Although Pat's beliefs were garnered directly from audio-taped protocols, there is no foolproof algorithm for translating utterances into propositions, so analysis has some latitude. Similarly, although we tried to include only relations that were explicitly used in Pat's explanations, this part of the analysis also involves some subjectivity. It is particularly difficult for the coder to refrain from adding an obvious node or a link even though the particular subject didn't vocalize that obvious belief or relationship. (For instance, the authors found it difficult to *not* add an inhibitory link between Pat's AH2 and NH2.) Constructing Hal's belief system allowed for more latitude than Pat's, since he is a composite. Still, care was taken to create the network first -- before tinkering with the processing parameters -- so that we would be less likely to "kludge" the representation.

There is also another kind of representational question: What does one of these networks actually represent? Generally, we conceive of the networks as models of the current contents of working memory. Note, however, that by "current" we also mean "contextual," because subjects can hold a belief in one context that they disbelieve in another. For instance, in an abstract context, most subjects explicitly held CP1, that there is no speed at a pendulum's endpoints -- even those, like Hal, who would reject it (in favor of AH1) during the context of the pendular-release tasks.

A general problem with connectionist cognitive models is that they usually have numerous numerical parameters that can be manipulated to produce desired results. Does our simulation depend on fine parameter tuning? The most important parameters in ECHO include the weight value of excitatory links, the negative weight value of inhibitory links, the weight value of the (data priority) links between evidence and the special evidence unit, and the decay of each unit at each cycle. The simulations of both Pat and Hal used the same parameter settings, and yielded the desired trajectories over similar ranges for each parameter. These common parameter ranges were: .015 to .05 for excitatory weights, -.05 to -.065 for inhibitory weights, .035 to .075 for data-priority

weights, and .01 to .065 for the decay rate.

The simulations *might* have employed even more parameters. For instance, we treated units representing direct observations, memories, and facts all as evidence, with each linked to the special evidence unit by the same weight. But one can argue for varying these weights for different kinds of evidence, increasing them for current observations and decreasing them for fuzzy memories. Not all evidence has the same epistemic status. In particular, when Hal is directly presented with a phenomenon on the computer screen in front of him, this becomes a very salient piece of evidence. Accordingly, one might argue that the unit representing the surprising observation that the pendulum bob falls straight down at the end of its swing should be a multiple of the data priority of remembered evidence.

FUTURE RESEARCH

We have been modeling previously performed experiments, but ECHO can also be used to make predictions about the beliefs of subjects. Our simulation of Hal predicted that he would come to doubt the belief that kids can fly off the end of a playground swing, but very few subjects *explicitly* re-evaluated this belief. ECHO predicts that the subjects may have experienced this belief change even if they did not mention it, and this prediction can be tested in new experiments by asking subjects to state their confidence in belief E1 following the relevant feedback. Additional experimental tests of the extent to which ECHO models human performance can be done in situations where people face difficult inference problems involving judgments of explanatory coherence. We conjecture that problems that are relatively hard for people, as measured perhaps by the length of time to generate answers, will also be relatively hard for ECHO, as measured by the number of cycles it takes the system to reach a stable state. Legal reasoning, in which jurors attempt to construct a coherent account of the evidence (Pennington & Hastie, 1987), appears to be a particularly promising domain for future empirical tests of the ECHO model.

ACKNOWLEDGMENTS

This research was supported by a grant from the James S. McDonnell Foundation to Princeton University and by grant NR667-534 from the Office of Naval Research. We are also indebted to Gilbert Harman, Stephen Hanson, Lauren Resnick, and members of a discussion group on explanatory coherence.

REFERENCES

Caramazza, A., McCloskey, M., Green, B. (1981). Naive beliefs in "sophisticated" subjects: misconceptions about trajectories of objects. flCognition, 9,, 117-123.

Halloun, I. & Hestenes, D. (1985). Common sense concepts about motion. *American Journal of Physics, 53*, 1056-1065.

Harman, G. (1986). *Change in View.* Cambridge, MA: MIT Press.

McCloskey, M., Washburn, A., & Felch, L. (1983). Intuitive physics: The straight-down belief and its origin. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 9*, 636-649.

Pennington, N. & Hastie, R. (1987). Explanation-based decision making. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, 682-690.

Ranney, M. (1987a) *Changing Naive Conceptions of Motion.* Doctoral dissertation, University of Pittsburgh, Learning Research and Development Center.

Ranney, M. (1987b, April). *Restructuring Conceptions of Motion in Physics-Naive Students.* Paper presented at the annual meeting of the American Educational Research Association, Washington, DC.

Rumelhart D., & McClelland, J. (Eds.). (1986). *Parallel Distributed Processing.* (Vols. 1 & 2). Cambridge, MA: MIT Press.

Thagard, P. (1988a). *Explanatory coherence.* Princeton University Cognitive Science Laboratory Technical Report. Princeton, NJ.

Thagard, P. (1988b). *Computational Philosophy of Science.* Cambridge, MA: MIT Press/Bradford Books.

Thagard, P., & Nowak, G. (1988). *The explanatory coherence of continental drift.* Manuscript submitted for publication.

Wertheimer, M. (1945). *Productive Thinking.* New York: Harper.

Access and Use of Previous Solutions In A Problem Solving Situation

Jeremiah M. Faries
Brian J. Reiser

Cognitive Science Laboratory
Princeton University

Abstract: An important component of problem solving is the ability to make use of previous examples. This requires noticing the relevance between the current and previous problems. We examine the role of the superficial and structural relations among problems and the remindings that these similarities elicit in a problem solving situation. Students learned to program in an electronic book environment in which they were able to store and later retrieve solved problems. Their use of previous solutions suggests that novices are indeed sensitive to structural similarities and can use retrieved solutions in new problem situations.

Most models of problem solving assume that well learned cognitive skills rely on procedures that have been generalized through use to apply to a variety of problems that may differ on irrelevant features (e.g., Anderson, 1983; Laird, Rosenbloom, & Newell, 1986). Before these procedures have been sufficiently generalized and tuned, however, problem solvers in the early stages of learning a domain may rely on retrieving a previously solved problem and then modifying the solution to fit a current problem (LeFevre, 1987; Ross, 1984; Pirolli & Anderson, 1985). In contrast to procedural models of problem solving, expertise has also been characterized as the ability to access and modify previous solutions to fit new problems (e.g., Kolodner, 1987; Hammond, 1986; Ashley & Rissland, 1987). In these case-based reasoning models, even experts in a domain may rely on modifying a previous solution rather than applying general procedures.

Novices in a domain present an interesting challenge for models of case-based reasoning. Learning about a domain provides the knowledge to organize and encode the features of a problem important to the structure of its solution, so that it can be retrieved later in appropriate circumstances. Experts, however, may need to rely less on remindings than novices, who have not yet formed generalized procedures. The ability to retrieve relevant previous solutions is particularly crucial for novices in a domain. Yet novices may be misled by unimportant superficial similarities between the problem under current consideration and previously encountered problems.

Several recent studies have examined the conditions under which case-based remindings are likely to occur. Some studies have shown that the surface features of a given problem or story have a major influence on the possibility of being reminded (e.g., Ratterman & Gentner, 1987; Ross, 1987). Ross found that the superficial characteristics of word problems may play a major role in the memory processes that lead to these kinds of remindings, at least for novices. Similarly, Ratterman and Gentner found that superficial similarities between stories rather than structural similarities accounted for the large majority of cases in which subjects reported that one story reminded them of another story. One interpretation of these findings is that novices lack the knowledge to appropriately identify and/or categorize problems in a way that would foster appropriate structurally based remindings.

Indeed, structural and surface features are often correlated in the real world (Ratterman & Gentner, 1987) so accessing previous cases by relying on superficial characteristics is arguably a reasonable strategy for generating possible analogues, particularly if the novice lacked the ability to identify the structural components of a problem. It is not clear, however, that most novices are restricted to a strategy that reduces to "a shot in the dark". While it may be true that novices lack concepts and knowledge required to accurately categorize types of problems, it may not be true that they, therefore, rely on superficial characteristics to encode solutions in memory. Even novices may have enough knowledge to partially identify important features of the problem based on the problem solving goals, and should, in principle, be able to use these features to probe their own memories for related problems. This is not to say that surface characteristics of a problem will be unimportant in a reminding episode or that they won't be involved in retrieval in some way. Rather, the claim is that remindings in a problem solving situation will be more often guided by structural characteristics than by surface characteristics.

Hammond (1986) has criticized Gentner's argument on the basis that subjects in these experiments were not given specific problem solving goals. Understanding a story may not provide a well defined goal setting to encourage the use of structurally based remindings. There was no requirement to use the information in these stories, e.g., to make predictions about a current story or solve some type of problem, so it is difficult to specify the clearly structural aspects of the stories or the clearly superficial aspects. Indeed, Seifert, McKoon, Abelson and Ratcliff (1986) have demonstrated the importance of task goals in eliciting remindings.

Ratterman & Gentner's findings may be partly explained by the uncertainty the subjects may have had regarding the importance of the features they were using as a basis of the reported remindings. In a problem solving task where the nature of the task and goals are well-defined, the so-called superficial aspects of the problem are peripheral to the solution, hence it seems likely that remindings will infrequently be guided by superficial similarity alone. The motivation for the present research is to provide a problem solving situation within which structural remindings will be clearly useful, and in which the relevant aspects of the problems are clear. In this way it will be possible to assess the abilities of novices to notice similarities of varying degrees of structural correspondence and to assess the importance of superficial characteristics for access. The well defined nature of the task allows us to clearly determine which aspects of the problem descriptions may be superficial and which may be structural.

The present study is designed to examine remindings of previous solutions in a problem solving situation to test out the propensity and abilities of subjects to notice and make use of correspondences between problems. Subjects in this experiment learn to solve computer programming problems in an electronic book environment. Each problem they solve is stored away on the computer. Students can specify when they want to see any of their previous solutions to help them solve a current problem. By providing the opportunity for the subject to benefit from noticing correspondences, and by using an electronic book environment to trace problem solving and problem retrieval activities, we can investigate the kinds of similarities between problems that subjects noticed and decided to act upon.

THE BATBOOK ENVIRONMENT

We designed an electronic book environment, the Behavioral Analogy Tracing Environment (BATBook), to monitor subjects' use of previous solutions. In this environment, subjects read a text book on the computer screen, compose a solution in an editor, test the solution in an interpreter, and then file away the solution. The subject can request to see a previous solution at any time. In addition, it is possible to search the record of the interactions with the interpreter, and to search through the book for a particular target. The BATBook environment makes explicit a large proportion of the student's problem solving behavior, thereby providing a rich record for analyzing when and how subjects access the written instruction and their previous work.

In the present experiment, novices used BATBook to read the first two chapters of *Essential LISP*, a text book on the LISP programming language (Anderson, Corbett, & Reiser, 1987). Each chapter of the text contains several short sections of instruction followed by problems that apply the knowledge introduced in the section. In addition to the regular problems in the text, we added a second set of problems to Chapter Two. These problems were designed so that each problem contained a cover story similar to a previous problem and was structurally similar to a different problem from the first half. We were interested in whether subjects would retrieve solutions from the first half of the chapter while working on the second half, and whether these retrievals would be governed by surface or structural similarities. In addition, we included two variations on the method for storing previous solutions. One group provided a verbal label for each solution they constructed and later could use this verbal label to retrieve a previous solution, while the other group did not label their solutions and could retrieve them only by referring to the problem description or content of the solution.

The BATBook environment runs on Sun workstations. The screen contains a Text Window, an Exercises Window, a LISP Interpreter Window, and a Problem Submission window that appears when the subject stores a completed solution (see Figure 1).

Reading the Text: The left half of the screen contains the Text Window, which displays approximately one page of text from the book. When the student finishes reading the current page, he or she selects the "page forward" button (labeled "+") using the mouse. Subjects can page backward (the "-" key) and can return to the first page of the chapter (the "Page 1" key). The design of the Text Window was loosely based on the Superbook electronic book environment (Remde, Egan, & Landauer, 1987). When subjects reach the last page of the section, they are instructed to begin working on the problems associated with that section, and are not permitted to page forward until they have completed that problem set.

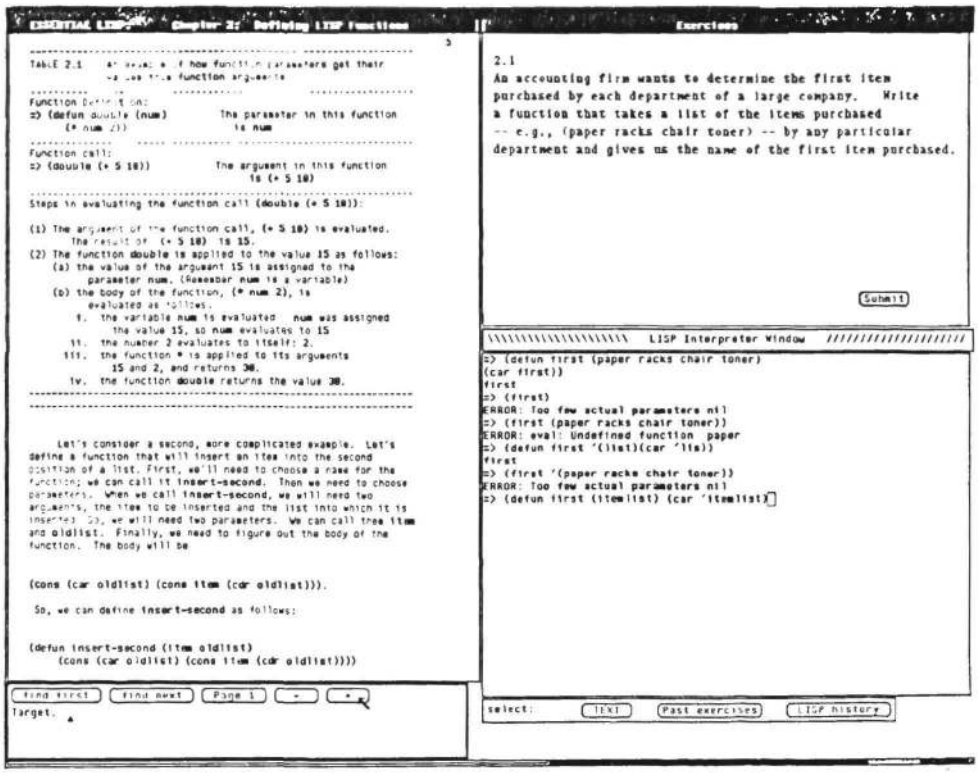


Fig 1. The BATbook screen layout showing text, problem description, and LISP work.

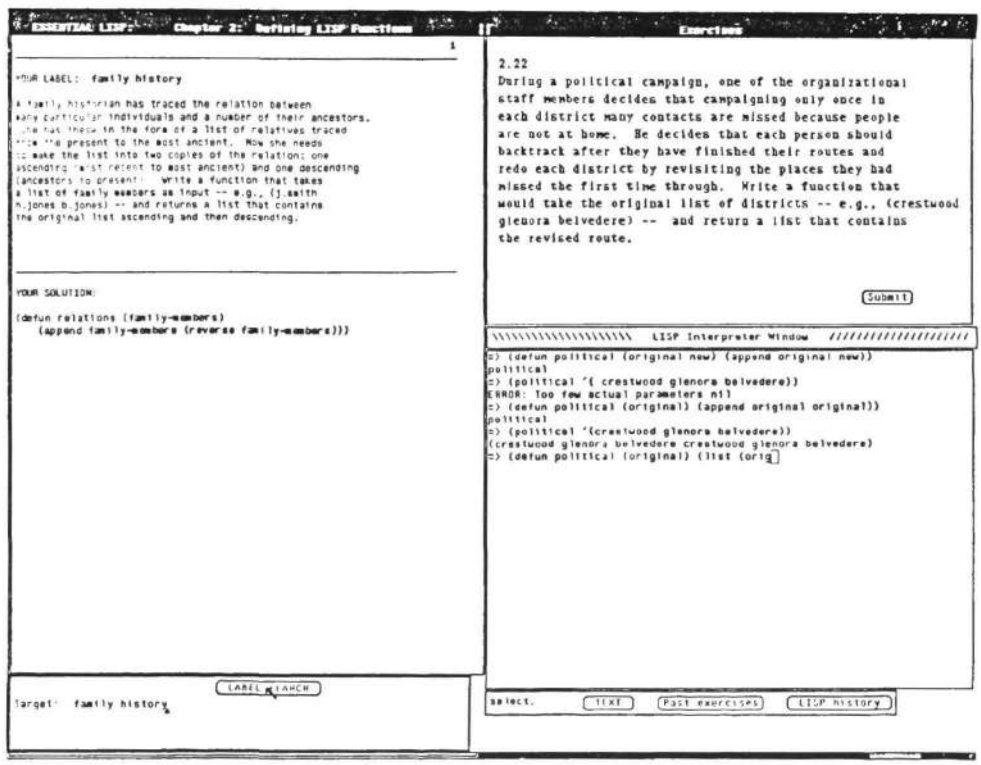


Fig 2. The retrieval of a previous solution using a label generated by the subject.

Searching the Text: Subjects can search through the current chapter of the text using a search capability similar to that provided in most word processing programs. To do this, the subject types a target of one or more words and then selects the "Find First" or "Find Next" key, which searches the text for the first occurrence or next occurrence of the search string. Thus, if the subject wishes to find a particular portion of text, he or she can page through the text or use the Text Search feature to find it.

Constructing a Solution: When the subject completed the reading for a section, he or she was instructed by the program to begin the associated problems. The problem sequence was initiated by clicking a "Start Problems" button in the Exercises Window, upon which the first problem for the section was displayed in the window. Problems required writing LISP function calls (Chapter One), and defining functions (Chapter Two). Subjects typed their function calls and function definitions into the LISP Interpreter Window which contained a Common LISP interpreter. Subjects could change a function definition by using a simplified Emacs editor invoked from the LISP Window. The editor contained commands to move left and right one character, up and down one line, delete a character, delete a line, and insert a deleted line. The editor also contained a commands to save the function and enter the revised definition into LISP, and to abort the edit with the definition unaltered. Typically subjects typed a definition into the LISP Window, tried their function on some examples, then edited the definition until it worked properly.

Submitting a Completed Problem: When the subject considered the solution to be correct, he or she submitted the problem in order to store the solution. This was initiated by clicking the "Submit" button in the Exercises Window. At that point, BATBook prompted the student to enter the function name and then checked the student's solution. If the solution was incorrect, the student was informed of the example for which their function computed an incorrect result and asked to try to fix their solution. Subjects were required to attempt to fix their function and submit a second solution. If the solution was still incorrect, the subject was again informed but this time was given the option to continue with the next problem, or to continue trying to fix the solution.

Labeling the Solution: One group of subjects (Label Group) was asked to label their final submitted solution (whether correct or incorrect) for the problem. The subject was asked to type in a brief label for the solution. Subjects were told that the label could be used at a later time to retrieve the problem description and the solution. The other group of subjects (Non-Label Group) were not asked to label the problem, and could later refer to the problem using a probe from either the problem description or the content of the solution itself. Following the completed problem, a new problem was displayed in the window. Upon completion of the problem set the subject was instructed to resume reading the text.

Searching Previous Solutions Using Labels: When a subject in the Label group wanted to look at a previous solution to help with a current problem, he or she clicked the "Past Exercises" key in the LISP Interpreter Window. At that point, subjects typed in part or all of a previous label and clicked the "Label Search" key to initiate the search. If a problem with a matching label was found, the complete problem description and the subject's final solution to the problem were displayed in the Text Window (see Figure 2). After finding a previous solution, the subject typically returned to the LISP Interpreter Window to attempt to map something from the solution to the current problem, or decided to search for a different problem if the search retrieved something that the subject then decided would not be useful. Subjects were informed that they might find it useful to search for previous solutions. There was no limit on the amount search that the subjects could do.

Searching Previous Solutions Using Keywords: When a subject in the Non-Label Group wanted to search for a previous solution, he or she clicked the "Past Exercises" button, and was prompted to type in one or more words to use as a search target. The subject then clicked the "Find First" or "Find Next" as in search of the Text Window. The first or next occurrence of the search string was sought among the problem descriptions and their solutions. If a match was found, the problem description and the subject's solution was displayed in the Text Window, just as for a Label Search. Only the problem and description for a single problem was displayed at a time.

Searching the LISP Interaction History: In addition to searching the text or previous solutions, subjects could also search the log of their interaction with the LISP Interpreter. We included this feature because we expected that subjects might find it useful to retrieve an episode of testing or debugging a function, in addition to retrieving their final answer to a problem. Search of the LISP history was initiated by selecting the "LISP History" button, whereupon subjects were prompted to type their search string as in the Keyword search. The search found the first or next occurrence of the search string in the log of the student's interaction with the LISP interpreter for that chapter. The log was displayed in the Text Window

with target string in inverse video.

At the conclusion of each problem, the display of the Text Window returned to the page that instructed them to continue with the problem set so that previous solutions, LISP history, or text would no longer be displayed. The subject would be required to search again to retrieve desired information.

Procedure: Subjects were run individually in three two hour sessions. The last two sessions were always on consecutive days. Subjects were free to explore any available aspect of the environment during the session. The LISP Interpreter was available throughout the entire session. The system required the subject to read each section and submit the associated problems before going on to next section. Subjects worked on Chapter One in the first session, and on Chapter Two in the second session. The third session concerned the "review" problems, which contained surface and structural similarities to the earlier problems in Chapter Two. No additional reading was presented in the third session.

Materials: The problems of Chapter One were included unchanged from the *Essential LISP* text. Cover stories were added to the 11 original questions of Chapter Two, the "source" problems. Each question was constructed using a different cover story. These cover stories were plausible scenarios for the problems, but were unrelated to the structure of the solution. The third session contained the 11 "target" problems. Each target problem was constructed so that it contained the same cover story as one of the source problems, and was structurally similar to a different source problem. Thus, it would be possible for each target problem to retrieve a previous problem focusing either on surface or on structural similarity. An example target problem and its surface and structural source problems is shown here:

Target: During a political campaign, one of the organizational staff members decides that when campaigning only once in each district many contacts are missed because people are not at home. He decides that each person should backtrack after they have finished their routes and redo each district by revisiting the places they had missed the first time through. Write a function that would take the original list of districts -- e.g., (crestwood glenora belvedere) -- and return a list that contains the revised route.

Structural similarity source: A family historian has traced the relation between many particular individuals and a number of their ancestors. She has these in the form of a list of relatives traced from the present to the most ancient. Now she needs to make the list into two copies of the relation: one ascending (most recent to most ancient) and one descending (ancestors to present). Write a function that takes a list of family members as input -- e.g., (j.smith h.jones b.jones) -- and returns a list that contains the original list ascending and then descending.

Surface similarity source: A political campaign organizer is making up lists of neighborhoods for the campaign workers to visit. He would like to have his workers visit the districts at different times of the day so that people who are not home at particular times of the day may be reached on another day. To do this he plans to have his campaign workers first visit the district they visited last on the day before, and then continue in the same order as they did previously. Write a function that takes a list of district names -- e.g., (crestwood glenora belvedere) -- and returns a list with the last district name moved to the beginning.

Subjects: Subjects were 10 Princeton University Students, community members, and research staff, who were paid $5 per hour for their participation. Subjects were selected who had no formal training in a computer language and very little or no informal programming experience. The subjects were semi-randomly placed in each condition with consideration given to extreme math SAT scores in order to balance these individuals between conditions. The average SAT score was 670 in the Label condition and 690 in the Non-Label condition.

RESULTS AND DISCUSSION

Throughout all three sessions there were many cases in which subjects referred to previous problems, past work in the LISP History, and previous portions of the text. We will focus our analyses on the search for previous work during the review half of Chapter Two (day 3). It was in this set of problems that every problem had one structural and one superficial counterpart in the initial set. There were an average of 6.4 cases of searching behavior in Chapter One and 5.3 cases in the first half of Chapter Two.

Type of Problem Retrieved: Subjects in the review half of Chapter Two exhibited an average of 6.5 cases of search behavior. We categorized the searches of solutions and LISP History according to whether the retrieved problem was the superficial antecedent, structural antecedent, or a problem unrelated in the design. Of the total searches, 49% retrieved a structurally similar problems while only 9% retrieved a problem with superficial similarities. The large difference supports our claim that novices are indeed sensitive to structural correspondences, and can exploit this recognition to use their solutions in later problems. The remaining 42% of the retrieved problems were of unspecificed similarity.

These include cases in which the subject rejected the retrieved problem and searched again with a different cue, and cases in which their solution or the errors they encountered were similar in ways not captured in the design.

Considering the two groups separately, the Label subjects exhibited an average of 10.4 remindings. Of these searches, 57.7% retrieved structural counterparts, 5.8% retrieved superficially related problems, and 36.5% were unspecified. Subjects in the Non-Label condition exhibited far fewer instances of search behavior, only 2.6 cases per subject. Of these only 15% represented searches for structurally related problems, 23% represented searches for superficially related problems, and 62% represented searches for unspecified problems.

It is important to consider the relations between those retrieved problems that were neither structural nor surface antecedents. There may have been structural correspondences between problems or the difficulties encountered in the solution other than those between problems designed to be structurally similar. Therefore, we considered the events that precipitated the searches for previous solutions. Interestingly, 78% of the structural remindings occurred immediately after reading the problem description. 19% were precipitated when the subject encountered an error, and 3% occurred after trying LISP calls prior to attempting a function definition. For surface related searches, 50% followed problem descriptions, 33% followed errors and 17% occurred after the problem was solved. Of the unclassified searches, 41% followed problem descriptions and thus provided no clear indication of the reason for the search. An additional 37% of the searches were due to extensive trial and error searching by one subject who had used numbers as his solution labels. Finally, 22% of the unclassified searches followed errors, and almost always retrieved a previous portion of the LISP History in which a similar error occurred. Thus, these searches that retrieved "unrelated" problems in fact recovered solutions to problems that differed on superficial features but bore an important structural similarity in the types of difficulties encountered in constructing the solution. Interestingly, none of the subjects whose search was preceded by an error actually used part of the error message to find a corresponding part of the history. Instead, these typically used a function or variable name related to superficial characteristics of the problem.

Type of Search Key Used: Subjects were free to construct their own probe words for the search. We categorized the search probes used by the subjects into those that referred to structural and surface features of the problem. Surprisingly, only 23% of the probes were related to the structural features of the problem, whereas 77% were related to superficial characteristics. The environment may have biased the use of superficial features, since the most distinctive strings in the targets were usually related to the superficial aspects of the problem. Nevertheless, it is important to stress that subjects could often locate a structural correspondent using a superficial aspect of the event. For example, one subject, after reading the target problem description shown in Figure 2, searched his LISP history for the word "family" which was the name of the function he had defined on the previous day for the structurally isomorphic problem.

The surface information is evidently accessible and can be used to identify structural correspondences. The amount of structurally based search suggests that people organize memory for events in ways that reflect structural and functional importance. The use of superficial search keys suggests that superficial information is retained, although it appears that access to this information was through structural routes. It is even possible that superficial information helps to to cohere the structural aspects of the memory. Although it might be argued that subjects remembered events using surface features as cues and them made assessments of structural soundness to decide whether to use the event, the short latencies between problem presentation and initiation of structural searches make this alternative unlikely.

We also considered the proportion of searches in which subjects successfully utilized the retrieved information. Success was defined as a correct solution following a search; this occurred on 62% of the searches. Failures included cases in which retrieving a previous solution led to an incorrect solution to the current problem, and those cases in which the retrieved information was rejected in order to initiate another search. These comprised 38% of the searches. A structural probe led to a successful search 67% of the time, whereas it led to a failure 33% of the time. A superficial probe led to a successful search 57% of the time, whereas it led to a failure 43% of the time. The failures also include behavior such as using trivial labels (one, two, three) to "page" through the problem histories. This occurred in two episodes and accounted for 39% of the so-called failures. It should also be noted that many of the failures formed part of episodes that led to a successful search in a few steps.

There results suggest that the nature of the probe used did not affect the probability of successfully retrieving useful information. Surface features

can be used to retrieve structurally relevant episodes, even though they are not structurally diagnostic. They are a part of the memory for the event that is salient, but not necessarily influential in the selection of isomorphs. We also considered the type of labels, function names, and variable names generated by subjects. Here again, many of these labels and function names referred to surface features of the problem, yet this did not prevent these labels and solution components from being effective search cues to retrieve structurally related problems.

Conclusions: We have presented evidence that people in a clearly defined problem solving situation are sensitive to and are able to make use of structural correspondences between problems. The subjects were rarely misled by the superficial correspondences and were able to identify and make use of structural similarities between problems. They may not have had all the well formulated rules they needed, but they were sensitive to the structural nature of the problems and could, therefore, detect functional similarities. The efficient use of examples requires subjects to have encoded the relevance of particular examples, and to remember enough about them to generate successful retrieval descriptions.

As Ratterman and Gentner (1987) have demonstrated, superficial characteristics may be important for access, but this may not necessarily reflect how novices store information for previous problems. The availability of superficial information does not imply that the structural information is unavailable. In fact, it appears that the structural organization of the memories for the problems solved may have been responsible for the activation of the surface level features. The use of superficial probes to locate structurally similar problems demonstrates that surface features are not dissociated from information used to make inferences between related problems. Surface features may simply be more salient and easier to specify than more abstract features. This salience does not necessarily interfere with problem oriented memory organization and retrieval.

ACKNOWLEDGEMENTS

We are grateful to Eric Ho and Antonio Romero for assistance in programming BATBook, and to Dennis Egan, Louis Gomez, Tom Landauer and Joel Remde of Bellcore for discussions of their Superbook program. The research reported here was supported in part by contract MDA903-87-K-0652 from the Army Research Institute, and by a research grant from the James S. McDonnell Foundation to Princeton University. 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

Anderson (1983). *The architecture of cognition.* Cambridge, Mass.: Harvard Univ. Press.

Anderson, J. R., Corbett, A. T., & Reiser, B. J. (1987). *Essential LISP.* Addison-Wesley.

Ashley, K. D., & Rissland, E. L. (1987). Compare and contrast, a text of expertise. *Proceedings of AAAI-87, Sixth National Conference on Artificial Intelligence*, Seattle, WA.

Hammond, K. (1986). The use of remindings in planning. *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, Amherst, MA.

Kolodner, J. L. (1987). Extending problem solver capabilities through case-based inference. *Proceedings of the Fourth International Conference on Machine Learning*, Irvine, CA, Morgan Kaufman, 167-178.

Laird, J., Rosenbloom, P., & Newell, A. (1986). *Universal subgoaling and chunking* Boston, Mass.: Kluwer.

LeFevre, J. (1987). Processing instructional texts and examples. *Canadian Journal of Psychology*, 41, 351-364.

Pirolli, P. L., & Anderson, J. R. (1985). The role of learning from examples in the acquisition of recursive programming skills. *Canadian Journal of Psychology*, 39, 240-272.

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*. 23-35.

Remde, J. R., Gomez, L. M., & Landauer, T. K. (1987). Superbook: An automatic tool for information exploration - hypertext?. *Proceedings of Hypertext '87.*

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. *JEP: LMC*, **13**, 371-416.

Seifert, C. M., McKoon, G., Abelson, R. P., & Ratcliff, R. (1986). Memory connections between thematically similar episodes. *JEP: LMC*, **12**, 220-231.

The Use of Explanations for Completing and Correcting Causal Models[1]

Joel D. Martin and Michael Redmond
Georgia Institute of Technology
E-mail: joel@gatech.edu, redmond@gatech.edu

Abstract

Causal models describe some part of the world to allow an information system to perform complex tasks such as diagnosis. However, as many researchers have discovered, such models are rarely complete or consistent. As well, the world may change slightly, making a previously complete model incomplete. A computational theory of the use of causal models must allow for completion and correction in the face of new evidence. This paper discusses these issues with respect to the evolution of a causal model in a diagnosis task. The reasoner's goal is to diagnose a fault in a malfunctioning automobile, and it improves its diagnostic model by comparing it with an instructor's. A general process model is presented with two implementations. Related work in explanation based learning and in incorrect causal models is discussed.

Keywords: Learning, Causal Models, Explanations, Diagnosis

INTRODUCTION

A causal model or domain theory is an essential ingredient in understanding complex situations. For example, in order to diagnose a fault in a complex system, the diagnostician must be capable of making guesses about what might be wrong. However, without appropriate heuristic knowledge to guide and make those guesses, the correct hypothesis may never arise. Although many researchers have acknowledged the need for such causal models [Kuipers, 1984] [deKleer & Brown, 1981], very few have been concerned with the possibility that the domain theory may be incomplete or inconsistent. Those researchers who have recognized this problem [Rajamoney & DeJong, 1987] have not yet allowed for modification of the underlying causal theory.

With this in mind, Lancaster and Kolodner [1987] took protocols of the diagnostic behavior of novice, intermediate, advanced, and expert car mechanics. They observed evidence for a working model, a set of symptom fault pairings, and diagnostic strategies. They also observed [Lancaster, personal communication] that less experienced mechanics had inconsistent and incomplete knowledge, as one might expect. The research presented in this paper represents an effort to discover how an incomplete causal model (novice) can evolve to a more complete (experienced) state as a result of problem solving experience coupled with explanations about how those problems are solved.

Our model is implemented in two computer programs called EDSEL-1 and EDSEL-2 (Explanation in Diagnosis: the use of Symptoms, hypotheses, and Explanations for Learning) that each begin with a novice memory and are presented with problems and explanations of how to solve those problems. Specifically, the systems "watch" or attend to an instructor who is diagnosing a fault in an automobile. As they do so, they attempt

[1]This research was supported by the Army Research Institute for the Behavioral and Social Sciences under Contract No. MDA-903-86-C-173. The authors wish to thank Janet Kolodner for her advice and guidance, and Mark Graves and Hong Shinn for helpful comments on earlier versions of the paper.

to identify missing information of various types or to identify whether there is an inconsistency. If one of these problems is discovered, the systems modify their causal model to prevent the difficulty in the future. The recognition and modifications are based upon an attempt by the systems to explain their input.

The paper describes a general algorithm, presents the issues involved in completing and correcting causal models, and compares the two implementations.

GENERAL PROCESS

Completing and correcting a causal model requires a reasoner to recognize when it is missing a piece of information and then to incorporate that information into the model. This notion is complicated by the fact that there are different types of information in the model and that existing knowledge affects how new information is incorporated.

Redmond and Martin [1988] noted in the protocols from Lancaster and Kolodner [1987] that an instructor provides the students with a symptom, a series of hypotheses, and explanations for those hypotheses. We demonstrated that a system may process these inputs by attempting to build causal chains between hypotheses and the symptom, using provided explanations if no causal chain is obvious. If a complete chain can be built, it can be collapsed and be used more efficiently in future similar situations. If a complete chain cannot be built, then the instructor's explanation can be helpful, either by being added directly to the causal model, or by allowing the gap in the chain to be bridged. Our name for this process is Learning by Understanding Explanations (LBUE). The causal model contains frames [Minsky, 1975] for the components of a car, with slots for inputs, outputs, connections, parts, functions, and causal relationships between structures. For example, one piece of the current model is:

```
Starter:
   (isa component)                              ;A starter is a component.
   (part-of starting-system)                    ;Is a part of the starting system.
   (input electricity battery battery-cables)   ;Electricity from battery via cables.
   (parts starter-pinnion-gear starter-motor)   ;PARTS: pinnion gear and starter motor.
   (function spin-action starter-pinnion-gear)  ;FUNCTION: spin the pinnion gear.
   (cause (switch-action solenoid on)           ;Solenoid switch causes the two gears to interlock.
       (interlock starter-pinion-gear flywheel-ring-gear))
   (cause (crank starter-pinion-gear)           ;Cranking one gear causes the other to crank.
       (crank flywheel-ring-gear))
```

The causal chaining process uses the causal relationships and some of the related knowledge. Besides the causal model of the domain, the LBUE approach also includes a set of likely symptom-fault pairings, as observed by Lancaster and Kolodner [1987] . These sets of pairings associate a symptom with a problem, and are used to derive initial hypotheses during diagnosis, and to index into the causal model at the appropriate place. The general algorithm for the process is as follows:

1. From the symptom (presented by the instructor), chain backward, inferring possible findings that could lead to the symptom.
2. From each hypothesis (presented by the instructor), chain forward, inferring possible effects that could be caused by the hypothesized fault.
3. If the symptom chain meets a hypothesis chain, then the reasoner has an explanation for the hypothesis, and the generalization that (CAUSE HYPOTHESIS SYMPTOM) is added to the symptom fault table and to the causal model.

4. If the chains do not meet - the reasoner does not have enough information to explain the hypothesis. In this case, it uses an explanation presented by the instructor,
   (a) Chain backwards from the explanation toward the hypotheses chain.
   (b) Chain forward from the explanation toward the symptom chain.
   (c) If both directions can be linked, then the most general relationship (CAUSE HYPOTHESIS SYMPTOM) can be learned.
5. Add explanation to the causal model.

The LBUE process results in an updated causal model. As discussed in following section, the things that may be learned are,

1. new objects
2. new relationships between objects
3. new causal information
4. new symptom fault knowledge

After learning, diagnosis is more efficient and more powerful for the same or similar problems, because the symptom-fault set can provide more reasonable hypotheses more quickly and the causal model is more capable of verifying an explanation. In addition, what the reasoner learns depends on what it already knows, since the reasoner's ability to chain back from the symptom and forward from the hypothesis is affected by the knowledge in the causal model. This means that the chains could meet given one version of the causal model, and have an unbridgable gap given another version.

The alternative to the LBUE approach is simply to remember symptom-hypothesis pairs. However, this would require a system to have already experienced a fault in order to diagnose it; no general knowledge is retained.

ISSUES FOR COMPLETING AND CORRECTING CAUSAL MODELS

As outlined above, a good diagnostic reasoner tries to explain why an hypothesis causes a symptom. It is this process that allows for the recognition of different types of missing information, and mediates the addition of knowledge to the causal model. The process of explaining hypotheses identifies missing information that might be useful for diagnosis because diagnosis is itself explanation, and hence requires the same information.

TYPES OF MISSING KNOWLEDGE

In general, a causal model may be missing many causal relations necessary for diagnosis. A reasoner will recognize that a causal relationship is missing if an explanation of a symptom cannot be formed, either while watching an instructor or while doing diagnosis. As well, there are situations in which an unknown causal relationship will be presented to the reasoner. Both possibilities are simple to detect, the former when causal chaining fails or no reasonable hypothesis is generated, and the latter, when the reasoner is actually told that something is missing.

Another form of knowledge whose absence is easily detected consists of referred-to facts. In other words, when an object or general relationship between objects is asserted, but is not known, then it is missing from the model. Somewhat more interesting are implied facts. The reasoner guesses it is missing an implied fact when a causal relationship is stated or implied by an instructor that the reasoner believes requires a mediating fact. For example, a reasoner may know,

```
(INTERLOCKED gear1 gear2) & (SPIN gear1 'clockwise) --> (SPIN gear2 'c-clockwise)
```

and an instructor may state,

```
(SPIN starter-gear 'clockwise) --> (SPIN flywheel-ring-gear 'c-clockwise)
```

From this, the reasoner will recognize that it is missing a fact (i.e., that the two gears are interlocked).

The final type of information that a reasoner may be missing is essentially efficiency information. The reasoner must be able to arrive at a reasonable or correct hypothesis quickly. If it cannot, the causal model must be modified to ensure timely and correct diagnoses in the future. The reasoner can recognize that it is missing this kind of information if it arrives at an incorrect hypothesis during diagnosis or if its hypotheses differ from the instructor's.

METHODS OF HANDLING INCOMPLETE KNOWLEDGE

An instructor's explanation of a given hypothesis can lead to information being added in three different ways. The explanation itself could be an unknown causal relationship which can be added to the model directly. For example, if the instructor explained

```
(cause (corroded battery-terminals) (not (connect battery battery-terminals)))
```

and this relationship was not associated with either battery or battery-terminals in the causal model, then it can be added there. A second way that the instructor's explanation can be used is to enable filling a gap in a causal chain. Either the explanation filled the gap, or it was a better cue to information that was not being accessed in the causal model. If the causal chain that can be built from the symptom (NOT (RUN ENGINE)) is:

```
(not (run engine)) --> (not (spin crankshaft)) -->
    (not (down-stroke cylinder)) --> (not (combustion cylinder))
```

and the causal chain that can be built from the associated hypothesis (NOT (MOVABLE BUTTERFLY-VALVE)) is:

```
(not (movable butterfly-valve)) --> (flow air carburetor low)
```

then there is a gap in the causal chain — the hypothesis is not fully explained. If the instructor provides the explanation that low air flow into the carburetor leads to a low air/gas mixture as the air passes the fuel float bowl then the following results:

```
(not (movable butterfly-valve)) --> (flow air carburetor low) -->
  (mix air gas less)  -->  (not (combustion cylinder)) -->
     (not (down-stroke cylinder)) --> (not (spin crankshaft)) --> (not (run engine))
```

Not only is the causal relationship given in the explanation used in filling the gap, but the relationship that (MIX AIR GAS LESS) causes (NOT (COMBUSTION CYLINDER)) is accessible when it hadn't previously been accessible, since the cue of carburetor is now available. Between the two, the gap has been filled.

The third way in which the instructor's explanation can be used is to infer a relationship that would fill a gap in a causal chain. If the explanation doesn't allow bridging the gap as discussed above, causal relationships which bridge the gap, which are implied by the expert instructor, can be inferred. The instructor implies that there is a causal relationship

between the hypothesis and symptom and that the explanation lies along this causal chain. Gaps will be filled with inferences if there is some general knowledge that indicates a cause is possible. For example, a cracked wire can cause low electricity because (a) wires conduct electricity, and (b) a conduit affects what it conducts. This would be given lower credibility than other learned relationships. There may be several plausible but inconsistent inferences that might fill a gap; the one chosen could depend on confirmation from a human observer.

Knowledge can be added to an incomplete causal model by inferring facts from a cause. This would occur as a result of the starter-gear ring-gear example mentioned above. In this case, the reasoner will infer that the starter gear and ring gear are interlocked.

In a sense, inefficiently represented knowledge is a type of incomplete knowledge. The information that is needed is in the causal model, but is not useful because it cannot be accessed, or it is given insufficient credibility, or it leads to slow processing. For instance, the explanation can allow the access of knowledge that couldn't previously be accessed. Additionally, filling a gap in a causal chain, as discussed above, is a way of dealing with some inefficient knowledge. This allows collapsing the chain into a single causal relationship, which can be used for more efficient processing. In collapsing the chains, the LBUE method has some similarities to Explanation-Based Learning (EBL) [Mitchell, Kellar, & Kedar-Cabelli, 1986] [DeJong & Mooney, 1986]. In order to allow for proper generalization of variables [DeJong & Mooney, 1986], a substitution list is kept that indicates to what categories each feature in the example was matched in order to instantiate the causal relationships. The collapsed chain then uses the most general category for a feature as the variable name in the antecedent or consequent of the new causal relationship.

INCONSISTENT KNOWLEDGE

Since new information is being added to the causal model, there is a possibility that a contradiction may occur. A few types of contradiction are possible. In one case, the same condition could be believed to cause contradictory effects such as:

```
  (corroded battery-terminals) --> (connect battery battery-terminals)
& (corroded battery-terminals) --> (not (connect battery battery-terminals))
```

Alternatively, a chain might be possible from known information, such that a condition indirectly causes a contradiction of the condition.

Contradictions may not be detected immediately, though, because the causal information is distributed throughout the causal model, and because an arbitrary amount of causal chaining may be necessary to detect the contradiction.

In cases where the new input is found to be inconsistent, the source of the information can be used to decide what to believe. Knowledge from the expert is given precedence over older information. Information that contradicts the expert is either removed or its strength is decreased, depending upon implementation.

IMPLEMENTATIONS

Implementation of the general model described above has followed two parallel paths. This decision was made because the research is exploratory, and therefore should generate several alternative approaches to the problem, and highlight different inconsistencies and difficulties with the model.

EDSEL-1 is based upon a simple active semantic net, similar to local connectionist models such as McClelland and Rumelhart's [1981] interactive activation model. EDSEL-2 uses an enhanced version of the causal model described in Allison [1987] . Although the process in both implementations closely follows the general model presented above, there are two significant differences. First, when a *causal gap* is present but no explanation fills that gap, EDSEL-1 uses generic knowledge about what affects what, whereas EDSEL-2 uses a less general but far simpler notion of filling gaps between recently proposed forward chains from hypotheses and backward chains from the symptom. The first method is a more flexible metric for evaluating whether a given *causal gap* should be filled, and therefore should lead to more reliable causal relationships. The second difference involves where causal information is stored and how it can be accessed. EDSEL-1 does not address the issue of limited availability of causal relationships, whereas EDSEL-2 allows the more realistic situation in which causes are not maximally indexed when they enter the system. That is, they may not necessarily be retrieved when needed unless the proper cues are present. This is a more realistic and efficient approach for a system with a very large memory.

RELATED WORK

Rajamoney and DeJong [1987] has specifically addressed the problem of inconsistencies or missing information in a causal model for simulation. If more than one simulation is possible, his system will experimentally search for disambiguating features in the environment. Although this approach is clearly useful, it does not allow for modification of the general causal information in the model. It concentrates on quantitative values for the current situation, and does not learn any general knowledge.

As has already been noted, in order to update causal models, the current effort uses an explanation based technique that is in some ways similar to those of DeJong and Mooney [1986] and Mitchell et al. [1986] . Specifically, in diagnosis, a causal chain must be discovered in a potentially very large network of causal information. EBL can be profitably used to permit instruction to produce "short cuts" in that network.

Classical EBL, however, does not produce enough learning when the causal network is incomplete. This may be remedied by the learning by failing to explain (LBFE) [Hall, 1986] technique of isolating the information that is present in the input but is not understood, and subsequently adding it to the existing EBL system. Although the current model has not yet been described in exactly these terms, it is in fact an example of LBFE. It differs from Hall's work by proposing that the information that must be added in the absence of an explanation is not necessarily explicitly represented in the input. Also, the current effort presents a domain independent notion of LBFE that describes how potentially any diagnostic causal net might grow, whereas Hall's effort was, in his own view, domain specific.

One of the methods that is used to augment incomplete networks in the current approach is to use relationships that are more general than causes in order to infer causation. For example, an action and a state change that relate to the same object tend to be causally related. This technique was originally used by Pazzani [1987] and a similar approach was suggested by Russell [1987] .

CONCLUSIONS AND FUTURE DIRECTIONS

This paper has discussed the LBUE paradigm for dealing with an incomplete or inconsistent causal model in the training of car mechanics. The main contribution of the current effort is in the ability to accept new knowledge and incorporate it into the causal model, while using it to understand an explanation and form a new generalization.

There are several directions for future research. First, in diagnosis, causal chaining is not the only strategy used, though it was the most common in the protocols. The explanations used by the instructor reflect several different strategies. For this reason, and to allow strategies to be learned and improved, diagnostic strategies must be explicitly represented. Some initial work has been done on this representation.

Second, better representation of the causal knowledge is needed to take full advantage of the inferencing possible from qualitative models. The aim is to use more levels of abstraction to allow reasoning at whatever level may be appropriate. Third, more learning may be possible in this paradigm if Case-Based Reasoning [Kolodner & Simpson, 1984], a method of using previous episodes and evaluation of their results to suggest solutions to new problems, could be integrated.

REFERENCES

Allison, K. R. (1987). Use of a working model in fault diagnosis. In *Proceedings of the 25th Annual Conference of the Southeast Region ACM.*

DeJong, G. & Mooney, R. (1986). Explanation based learning: an alternative view. *Machine Learning, 1*, 145–176.

de Kleer, J. & Brown, J. S. (1981). Mental models of physical mechanisms and their acquisition. In J. R. Anderson (Ed.), *Cognitive Skills and Their Acquisition.* Hillsdale, NJ: Lawrence Erlbaum.

Hall, R. (1986). Learning by failing to explain. In *Proceedings of the National Conference on Artificial Intelligence.*

Kolodner, J. & Simpson, R. (1984). A case for case-based reasoning. In *Proceedings of the Sixth Annual Conference of the Cognitive Science Society.*

Kuipers, J. (1984). Commonsense reasoning about causality: deriving behavior from structure. *Artificial Intelligence, 24*, 169–203.

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.*

Lancaster, J. (personal communication). August, 1987.

McClelland, J. L. & Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception: part 1. an account of basic findings. *Psychological Review, 88*, 375–407.

Minsky, M. (1975). A framework for representing knowledge. In P. H. Winston (Ed.), *The Psychology of Computer Vision.* New York: McGraw-Hill.

Mitchell, T. M., Kellar, R. M., & Kedar-Cabelli, S. T. (1986). Explanation based learning: an unifying view. *Machine Learning, 1*, 47–80.

Pazzani, M. (1987). Inducing causal and social theories: a prerequisite for explanation-based learning. In *Proceedings of the Fourth Annual International Workshop on Machine Learning.*

Rajamoney, S. A. & DeJong, G. F. (1987). *Active ambiguity reduction: An experimental design approach to tractable qualitative reasoning.* Technical Report UILU-ENG-87-2225, University of Illinois at Urbana-Champaign.

Redmond, M. & Martin, J. (1988). Learning by understanding explanations. In *Proceedings of the 26th Annual Conference of the Southeast Region ACM.*

Russell, S. J. (1987). Analogy and single-instance generalization. In *Proceedings of the Fourth Annual International Workshop on Machine Learning.*

Varieties of Learning from Problem Solving Experience*

Juliana Lancaster
BDM Corporation
1300 N. 17th St., Suite 950
Arlington, VA 22209

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

March 30, 1988

1 Introduction

Problem solving ability appears to be acquired in two ways. Some of the knowledge required for problem solving can be obtained through direct instruction or "book learning", but major portions of what experts know is derived directly from their experiences with problems in the domain of interest. As work in artificial intelligence has begun to focus on highly complex domains, understanding the contribution of experience to knowledge and behavior has become increasingly important.

As a result, researchers have begun to consider how experience leads to new knowledge and, specifically, the role of single cases in learning. In earlier work on psychiatric diagnosis, Kolodner (1982) developed some proposals about how an individual problem solving experience can lead to changes in the knowledge base of the problem solver. That research focussed on the changes in knowledge that can result from a diagnosis that was initially incorrect and then corrected. We argued that an analysis of the knowledge used in making a new, correct diagnosis would lead to the individual becoming aware of the errors or missing knowledge in his knowledge base that led to the original incorrect diagnosis. At that point, the problem solver would modify his knowledge appropriately to ensure that the same error would not recur. The modification process had two parts to it: modification of the problem solver's general knowledge and problem solving procedures and indexing of the experience in memory so that it could be remembered during later problem solving.

Our goal in the study reported here was to look at less experienced problem solvers to see what processes were involved in their learning. The processes involved in modifying general knowledge and problem solving procedures requires a fair amount of expertise on the part of the problem solver, both about the domain and about what could easily be done wrong in problem solving. Novices know very little about either of these things to do a full analysis themselves of their actions. We wanted to find out what kinds of learning procedures are available to them, what they can learn through problem solving experience, and what kinds of supervision are necessary for learning to happen. We also looked at which of their experiences they remembered to help them during later problem solving.

The domain we have been exploring is automotive diagnosis. This domain is of interest for several reasons. The automobile engine is a highly complex entity, consisting of several interacting subsystems. Failures in any component or subsystem produce symptomatic behaviors in engine performance, but the symptoms are not necessarily unique to a given failure. In other words, several failures can produce the same symptom(s). It is the mechanic's task to identify the correct failure.

*This research was supported in part by the Army Research Institute for the Behavioral Sciences under Contract No. MDA-903-86-C-173. Thanks to Phyllis Koton, Joel Martin, and Michael Redmond for commenting on an earlier draft of the paper.

Furthermore, the domain is knowledge-rich and both depth of knowledge and the ability to use it are important in reaching a good diagnosis. A student can be taught about cars and troubleshooting in general, but since cars vary extensively, and models may change significantly from year to year, no one can learn everything within an instructional setting. Clearly experience with different types and ages of cars is necessary for an individual to develop adequate expertise. An expert mechanic must have sufficient experience to allow him to draw his own generalizations about cars and to organize and access his knowledge in a manner appropriate to the specific car and problem he is considering.

In the work reported here, we observed the diagnostic process in several individuals and compared the knowledge used to solve successive similar cases. We also observed the interactions between problem solvers (students) and outside sources (books and instructors). The analysis presented here uses data presented in Lancaster & Kolodner (1987). That work identified the types of knowledge and diagnostic processes used by novice car mechanics.

Four automotive mechanics students at a local two year technical school were observed while diagnosing car failures. Six problems were presented at weekly intervals and think-aloud protocols were collected while the students worked. Each week, after all students had attempted to solve the problem, the instructor demonstrated the correct diagnostic procedures and his method of identifying the failure. Thus each student had a opportunity for feedback about his performance regardless of whether or not he diagnosed the problem correctly. Each failure was introduced into a car deliberately and each failure was caused by only one failed part. Four of the six problems presented the same symptoms initially, and a fifth problem presented a related symptom.

After all protocols were coded, two sequential protocols for each student were selected. The two protocols were selected to have the same initial symptom, but different failures. For each of these protocols, a description of the knowledge used and sought by the student during diagnosis was compiled. The descriptions of the two protocols were then compared and differences were noted. Of particular interest were occasions in which the student appeared to use knowledge in solving the second problem that he did not have in solving the first problem. Changes in the knowledge used showed us what students learned between solving the two cases. We then examined the protocols and the instructor's explanation of how to solve the first problem to see if we could identify how these items were learned. Sometimes this was obvious – a student had asked a question previously or the instructor had presented the relevant material in a previous explanation. Sometimes it was not obvious – what was learned might have been presented in class between the two exercises. We focussed on those learned items where we could identify the situation in which the new concepts were learned.

2 What was learned

As reported in our earlier paper (Lancaster & Kolodner, 1987), all of the students seemed to possess the same types of knowledge structures: a causal model of the engine, symptom-fault sets linking particular symptoms to the failures they might indicate, and a collection of troubleshooting guidelines and procedures. However, the students at various levels of training differed in both the amount of knowledge they had and the organization of that knowledge. The novice's knowledge was unorganized, sparse, and sometimes wrong. The intermediate student had a better organization on his knowledge (e.g., he grouped parts in the same system together) and some misconceptions had been corrected, but it was still incomplete. The advanced student had a much more complete knowledge base, and many of the misconceptions of the novice and intermediate students were corrected. In particular, the advanced student knew more about what failures normally look like than did the intermediate student, who knew more about how the car was supposed to work than about its malfunctions. The novice knew almost nothing about malfunctions.

The novice student, having learned the rudiments of how a car works in classroom instruction, was primarily focussed on acquiring the necessary organization of the causal model to allow

systematic tracking through the engine in search of a failure. For example, diagnosing a fuel system restriction in the earlier problem, the novice moved directly from the symptom (car cranks but will not start) to the hypothesis that it was not getting enough fuel and concluded that the fuel line was restricted. In a later case with the same symptom, the novice student instead showed that he had subsumed the fuel restriction hypothesis under a more global hypothesis of fuel system failure. He first checked the end point of the fuel system for evidence of failure and then began checking components within the system. Thus he had modified his knowledge base in an important way between cases. He had collected the fuel system components into a system, allowing him to consider the system as a whole as his first hypothesised failure source. As a consequence, his trouble- shooting behavior became more systematic as he only considered specific components of a system if the system as a whole was shown to be the source of the failure. This type of learning can be termed *knowledge reorganization.*

The novice also improved his problem solving skills. In the first case, he did not take action to confirm his diagnosis. Rather he was content to accept the hypothesis on the basis of its being a possible failure matched to the symptom. After attempting to solve the problem himself, he listened to the instructor's explanation of how to diagnose the a car's problem. The instructor, as part of his explanation, showed that it is necessary to confirm any hypotheses that are made. In the second case he solved, the novice had learned that confirming hypotheses is a necessary component of problem solving. He was thus also refining his problem solving skills.

In contrast, the intermediate student, having established the basic hierarchical organization of his causal model and symptom-fault sets, was primarily engaged in adding new information to his knowledge bases. Specifically, he showed evidence of adding to the faults associated with a given symptom in his symptom fault sets, learning new procedures for diagnosis, and identifying the locations and functions of new components. For example, on the earlier problem, the intermediate student made a protracted search for the fuel pump relay. On the later case, he was able to locate and test the relay at the appropriate time with ease. Thus, we see that he had learned not only its location but also how it was connected within the fuel system. Similarly, he employed techniques on the later problem that he had apparently not known when solving the earlier problem, leading to more efficient testing of hypotheses and an increased likelihood of getting correct, useful information from his testing.

The advanced student engaged in yet another type of learning that we have termed *refinement.* He already knew most of the information needed to successfully solve every problem presented. The changes in his knowlege between cases reflected primarily a reordering of faults in the symptom-faults sets to reflect new probablilties of occurrance. In other words, after solving a case in which the correct hypothesis was one he considered late in his troubleshooting procedures, the advanced student returned to that hypothesis earlier in his sequence on the later problem. He showed no notable changes in the overall organization or content of his knowledge between problems.

In addition, the advanced student seemed to have an additional model available to him that the intermediate and novice students were missing: a model of a *malfunctioning* car. Both the novice and intermediate student seemed to diagnose faults by first zeroing in on a system that might be faulty and then checking the behavior of the car against the expected behavior of a *working* car. What differentiated the novice's behavior from that of the intermediate student was that the novice might focus on a particular part, while the intermediate student focused on a system. The more advanced student (and the instructor), however, seemed to know what to expect in a *malfunctioning* car. In other words, he not only had a *working model* of the car, but he also had a *malfunction model* of the car. We can see this in his behavior when the testing equipment was faulty. The advanced student was the only one who could recognize that the readings he was getting from the test equipment were faulty. The intermediate student was unable to differentiate a bad reading on test equipment from a faulty car.

3 Learning Processes

Several learning processes were identified by examining the protocols:

Learning by understanding explanations: After the students had diagnosed each problem, the instructor demonstrated the correct, or optimal, diagnostic path for the problem. In doing so, he enumerated both the reasons for considering each hypothesis and the diagnostic and test procedures he was using. The students attempted to modify their knowledge to match the instructor's wherever his procedures, hypotheses, or explanations differed from theirs. We have termed this process *learning by understanding explanations* (Redmond & Martin, 1988, Martin & Redmond, 1988). It is a combination of learning by observing an expert (Mitchell, et al., 1985) and learning by being told. In learning by understanding, the learner has an opportunity to observe another person (in this study, the instructor) solving the same case while providing an ongoing explanation of the knowledge and processes used. The learner notices those points at which the instructor's knowledge differs from his own and modifies his knowledge to bring it into agreement with the instructor's. There are a variety of things that can be learned by this method: new organisational structures, as when the novice collected the fuel system components into a system; new diagnostic strategies, as when the novice learned to test system endpoints or when the intermediate student learned how to use trouble trees; new causal connections in the causal model; new test procedures; and refinements of any of these things.

A more complete explanation of this process can be found in (Redmond & Martin, 1988) or (Martin & Redmond, 1988). In short, a student learning by this method attempts to explain each of the things the teacher is doing, and then uses the teacher's additional comments and explanations to bridge the gaps in his understanding. Thus, if the teacher proposes a particular hypothesis given a particular set of symptoms, the student (internally) attempts to explain why that is a good hypothesis. The student may not know why a particular hypothesis is appropriate, but the teacher's explanation allows him to fill in the gaps in his attempted explanation. Without the teacher's explanations, the best the student can do is to draw associations.[1]

Active gap filling: While engaged in diagnosis, a student would sometimes realize that he did not know a specific piece of knowledge about the behavior of a component, or the connections between a component and the rest of the engine that was needed to solve the current problem. For example, his diagnostic procedures might have told him that he needed to check the input to some component, but he might not have known what the input source was. To find the missing knowledge, he would consult an outside reference source, such as a book or a more experienced individual. The information thus obtained was then incorporated into his knowledge base and was available during later diagnostic sessions. This is the process by which the intermediate student learned the location of and connections to the fuel pump relay. While he knew that he needed to find the fuel pump relay, he was unfamiliar with the particular car he was diagnosing. He knew that manuals provided this kind of information and went to the manual to find out. He went directly to the appropriate schematic in the manual and then spent considerable time using that schematic as a map of the engine and eventually finding the part he was looking for. In other words, he used an outside source (in this case, the manual) to find out exactly the specific piece of information he was missing.

We have termed this learning process *active gap filling* in recognition of the intentional nature of the learning. To engage in active gap filling, the individual must have enough knowledge about the system, enough problem solving knowledge to know what he ought to be doing, and an adequate organizational structure to allow him to recognize where the gaps in his knowledge might be, or at least to realize when an apparent dead end in his diagnosis might be the result of

[1] This process may remind people of Mitchell's (1985) learning by observing an expert, built into the LEAP system. Learning by understanding explanations assumes that the learner has incomplete knowledge and that the task of the learner is to augment its knowledge. LEAP's method, as well as other methods employing EBL and EBG (DeJong & Mooney, 1986, Mitchell, et al., 1986), assume that the learner has complete knowledge, and allow the learner to better package that knowledge.

a lack of easily obtained information rather than something more extensive. Possibly because of this requirement for a fairly complete and well organized knowledge base, active gap filling was primarily used by the intermediate and advanced students as a learning tool. Our protocols show that knowledge learned through active gap filling is remembered and available for use in later problem solving.

Learning from interpreting feedback: When a diagnostic procedure was not yielding the results the student expected, or when he could not interpret the results from a test, students had to ask for help. In essense, learning from interpreting feedback is a combination of active gap filling and learning by understanding explanations. As in learning by active gap filling, the student is aware of a gap in his knowledge. He may not, however, know what that gap is. And, while active gap filling is usually a process of finding out about some feature of an object, learning by interpreting feedback focuses on procedures: in particular, "what went wrong" with a particular procedure. Learning by interpreting feedback can result in correcting a faulty causal model, but more often involves correcting and refining knowledge about how to do things. This process is similar to learning by understanding explanations, since the instructor may provide a causal explanation of some phenomenon or offer advice about carrying out procedures. It is a more active and goal-directed process, however, initiated by some complication the student is experiencing while solving a problem.

Learning by interpreting feedback is invoked when the student cannot interpret the unexpected results of his problem solving. The process of explaining those results can be done by the student or by asking the instructor. When an intermediate student found that he was getting test results that he could not interpret he went to the instructor for help. On the same problem, the advanced student figured out for himself that the test equipment was faulty, and was able to learn on his own which ranges of test results predicted faulty equipment. The advanced student could do this, while the intermediate one could not, because he had knowledge telling him what things look like when they malfunction. This experience told him what malfunction of a particular instrument looks like. The intermediate student, on the other hand, did not know enough to hypothesize by himself that the test equipment might be faulty. On another occasion, however, the advanced student needed the instructor to provide an explanation to him. He was trying to energize the fuel pump from the battery using a test light to connect them together. He could not get the fuel pump to go on. He, like the intermediate student in the previous example, knew what the results should look like (in this case, he thought the fuel pump should go on) but did not know why the results he was getting were different. He asked the instructor what he was doing wrong. The instructor told him that he needed to use a lead (a wire with no bulb attached) to energize the fuel pump, since with a light attached, the bulb consumes the power from the energy before it gets to the pump. In this case, the advanced student's causal model of electricity was faulty.

Abstraction: Another learning process that was noted was *abstraction*. It seemed that any new information acquired by a student between problems was incorporated into the knowledge base at several different levels. We can see this most clearly in the novice student's behavior. Recall that in one problem, he tracked the failure to the fuel pump but did not examine the system further to distinguish whether the problem was in the fuel pump or in the input to the fuel pump. The instructor followed that session with an explanation of how to diagnose the fuel pump problem. In his explanation, he stated that the endpoints of the fuel system had to be checked for evidence of a problem. The student learned this (by understanding the instructor's explanation) and in the next problem applied it. The student also apparently learned the abstract principle the instructor was illustrating: that system endpoints need to be checked for problems when diagnosing any system. Abstraction is sometimes done spontaneously, as in the example just given, and is sometimes induced by the instructor during explanation. For example, in one instance the instructor walked students through an engine they hadn't seen before, and stated that parts with a particular function had to be found. He then explained what those parts look like in general. This abstract knowledge allows the student to identify the part no matter what type of car

he is looking at.

Case-Based Reasoning: A final process that facilitated learned was *case-based reasoning*. In case-based reasoning, a previous case that a problem solver is reminded of provides an answer to a new problem or focuses him on the knowledge needed to solve the problem (Kolodner & Simpson, 1984, Hammond, 1986). While in the previous paragraphs, we have referred to learning processes that allow a problem solver to learn new facts, case-based reasoning is a problem solving method that allows a problem solver to improve its performance without full understanding of the facts. Remembering previous cases allows a problem solver to solve a new problem better than an old one even when the problem solver is missing a causal explanation of why the previous solution did or did not work. Our hypotheses about case-based reasoning in experts predict that those cases that are different than what is expected are the ones that are remembered (Kolodner, 1982, 1983, Kolodner & Simpson, 1984, Schank, 1982). But novices don't always know the norms, so they can't recognize that something is different. The data we collected in this study show three situations in which novices stored cases for later problem solving. First, if a case serves as a strong justification for doing a procedure, then the case was kept and referred to later. The advanced student, in an early problem, spent 45 minutes trying to solve a problem that he could have solved instantly with visual inspection. In the next problem, he did a visual inspection immediately, saying "what we want to do is check the wire...like we did the last time". Second, cases were maintained in memory if they served as an example of a particularly complex or unusual situation. This includes, among other things, cases where a set of symptoms predict a highly unusual fault and cases where some set of symptoms predict a fairly commonplace fault that is hard to diagnose. Both students who solved problem 3 (one intermediate and one advanced) , in which a rocker arm had been removed, for example, were able to do it on the basis of remembering a previous case in which a rocker arm had been broken. A broken rocker arm is a highly unusual problem. In almost all cases, rocker arms outlive the cars they are in. The other students, who had never seen such a problem, were unable to solve the problem (one intermediate and one novice). These two students were able to determine that none of the possible common causes of the symptoms were responsible for the failure of the car, but they were not able to pursue their investigations beyond that point. The two successful students were reminded (based on the particular sound of the car) early in their diagnosis of two cars they had worked with in the previous quarter that had the same problem. This reminding led both of them to try the diagnostic procedures that would lead them to the correct diagnosis and, thus, to the diagnosis. A third instance in which a previous case was kept was when it illustrated something that was not known previous to that case. The case was remembered until what was learned from it was confirmed by a later case. While one of the other procedures might have been used to learn a general concept, the case in which it was learned seemed to remain available to the students until the new item of information was confirmed by another case. We could not determine, however, whether the general knowledge or the case was accessed first in later problem solving.

4 Discussion

While most researchers studying learning have been studying unsupervised learning or learning where the teacher gives an example but no explanations, our observations show that much of the learning that goes on early in problem solving depends on a teacher to give an explanation of the procedures for solving a problem and the knowledge needed to solve it. The causal model describing how a car functions properly, the one describing what malfunctions look like, associations between symptoms and faults, and the problem solving procedures the instructor finds useful motivate his explanations.

Early in learning, students don't know enough to be able to learn exclusively from their own problem solving experiences. For example, the novice who tracks down the error to the fuel pump when it is the fuel pump relay that is faulty (i.e., the fuel pump's imputs are causing it to malfunction, not the pump itself) can learn only if he is given extensive opportunity to attempt

to fix the problem or if a teacher intervenes to show him what he was doing wrong. And, when the student is a rank novice, the experience of trying to fix it himself may be so overwhelming that, in the worst case, nothing is learned, while in the best case, learning takes a long time. It might take several attempts, for example, trying to put in a new fuel pump before the student will start thinking that something else is the matter, and at that point, there are so many things that could have gone wrong during his previous attempts (e.g., the new fuel pump may be faulty, he may have connected it up wrong in a whole range of different ways) that he may only be able to track down the problem if he is lucky in his initial guesses.

Nor did our intermediate student seem capable of learning without supervision. While he didn't need help with simple diagnostic procedures, he was unable to differentiate between instrument readings produced by faulty parts and instrument readings produced by faulty instruments. In other words, his missing knowledge about what faults look like required that an instructor intervene to instruct him on that subject.

Not only did students need to know how things work (i.e., have a nearly complete *working model*) to learn in a completely unsupervised setting, but they also need to know how things can go wrong and what things are in the normal realm of possibility. The advanced student, who was by no means an expert, was able to do quite a bit more learning by himself because he knew both of these things. He also, however, needed help from time to time when the knowledge he needed to solve a problem by himself was missing or faulty.

We have attempted to outline the real-world learning procedures students use in learning a complex diagnostic task and the role of the teacher in learning such tasks. While much research has gone into unsupervised learning, little research has focused on processes by which a student learns from a teacher. Among the learning processes we have identified, three require extensive interaction with a teacher: learning by understanding explanations, active gap filling, and learning by interpreting feedback. The particular interactions depend on the knowledge the student already has. More research in this area is surely needed if we want to develop better teaching technologies and practices.

5 References

1. DeJong, G. & Mooney, R. (1986). Explanation-based learning: An alternative view. *Machine Learning*, Vol. 1.

2. Hammond, K. (1986). *Case-Based Reasoning: An integrated theory of planning,learning and memory.* Ph.D. Thesis. Yale University.

3. Kolodner, J. L. (1982). The Role of Experience in Development of Expertise. *Proceedings of AAAI-82.*

4. Kolodner, J. L. (1983). Maintaining Memory Organization in a Dynamic Memory. *Cognitive Science*, Vol. 7.

5. Kolodner, J. L. & Simpson, R. L. (1984). Experience and Problem Solving: A Framework. *Proceedings of the Sixth Annual Conference of the Cognitive Science Society.*

6. Lancaster, J. S. & Kolodner, J. L. (1987). Problem solving in a natural task as a function of experience. *Proceedings of the Ninth Annual Conference of the Cognitive Science Society".*

7. Martin, J. D. & Redmond, M. (1988). The Use of Explanations for Completing and Correcting Causal Models. Submitted to the 1988 Cognitive Science Conference.

8. Mitchell, T. M., Kellar, R. M. & Kedar-Cabelli, S. T. (1986). Explanation based learning: An unifying view. *Machine Learning*, Vol. 1.

9. Mitchell, T. M., Mahadevan, S., & Steinberg, L. I. (1985). LEAP: A Learning Apprentice for VLSI Design. *Proceedings of IJCAI-85.*

10. Redmond, M. & Martin, J. (1988). Learning by Understanding Explanations. *Proceedings of the 26th Annual Conference of the Southeast Region ACM*, Mobile, Alabama.

The Process of Learning LISP

Frederick G. Conrad and John R. Anderson

Department of Psychology

Carnegie-Mellon University

Modern research on skill acquisition is dominated by two broadly defined schools. One of these schools analyzes verbal protocols produced by a small number of subjects as they perform complex tasks in unconstrained situations. The other school measures the time and accuracy of larger numbers of subjects performing simplified tasks in controlled experiments. While protocol studies can provide rich detail about the acquisition of useful skills (Ericsson & Simon, 1985), the small numbers of subjects, the free ranging quality of their reports, and the theory laden interpretation of the data make reliability and generality unrealistic. The experimental approach, in contrast, yields precise, objective findings, but the distilled tasks given to subjects jeopardize the everday relevance of these findings (Neisser, 1976). In the current paper we report a series of analyses that exploit the power of both protocol and experimental methods while, we hope, transcending their limitations. We have tracked the acquisition of a complex skill, LISP programming, by examining student performance with the LISP Intelligent Tutoring System (Anderson & Reiser, 1985). This tool provides us with detailed, continuous records of student performance, much like protocol data; however, the nature of these data allows us to analyze them with the quantitative methods typically applied to experimental results. By converging these research methods, our goal is to present a more complete and accurate picture of learning LISP than is possible through either approach alone.

Students interact with the LISP tutor as they solve programming exercises in a one semester, introductory LISP course. Every exercise in the curriculum is decomposed into a sequence of production rules (Anderson, 1983), most of which correspond to typing LISP code. The student's input stream is segmented into separate productions when he or she types a delimiter key (space bar, carriage return or closing parenthesis). The input bracketed by each pair of delimiters is treated as the action of an underlying production rule. For each student action, the tutor tries to generate a corresponding piece of code by applying a relevant production rule. If the student and tutor code match, the student progresses through the problem. For each such match the tutor records the student's coding time, measured from delimiter to delimiter. A discrepancy between student and tutor code is recorded as the student's error and triggers a tutorial dialogue. The student then receives subsequent opportunities to produce the correct code. In the analyses that we report below, we restrict our discussion to latency measures. We further narrow our data by considering latencies from only those trials on which productions are executed without errors.

We are taking the production model in the LISP tutor as the simulation of each student and are using the LISP tutor to automatically analyze the student's protocol and determine a correspondence between that protocol and the production model. This paper will be concerned with how systematic the data appear under such an analysis. We expect that

students will improve as they go through the LISP tutor. To the extent that this improvement is systematic, this will both support the production model and provide evidence about the nature of skill acquisition.

A single LISP symbol (piece of code) can corrspond to several different production rules. What differs are the conditions on the rules. For example, one rule corresponds to *car* when used simply to retrieve the first element of a list; another rule is applied when *car* is called in conjunction with *cdr* to produce the second element of the list. The mean coding times for these rules in the first lesson are about 19 seconds and 45 seconds respectively. This example and many similar ones support the idea that psychologically, there is a fundamental difference between distinct rules with identical actions. While the mapping between surface behavior and underlying rule is usually one to one, the student must still acquire on the order of 200 production rules throughout the term. The various combinations of rules throughout the coding exercises produce about 3400 data points per student. We would like to identify the factors that influence performance at essentially each of these points.

Regression Analysis

We will restrict our analyses to 43 students who used the LISP tutor in courses taught in the fall of 1985 and spring of 1986. Because the data supplied by the tutor are continuous and quantitative, they lend themselves to analysis by multiple regression. Our goal is to formulate a regression model that allows us to predict coding latency at any point in the 125 problems that our students are required to solve. We considered a large number of potential predictors but only a few made independent contributions to predicting the variance We have been most successful in accounting for coding latencies using three predictor variables: (1) lesson number -- the material covered by the tutor is presented in 12 lessons. (2) log number of opportunities to code a production within a lesson, and (3) log absolute serial position of a production within a problem. Our criterion is actually log seconds. We chose a log scale for the dependent measure and two of the independent variables and a linear scale for the remaining independent variable because this combination of scales accounted for significantly more variance than other possible combinations.

We have subdivided our data into "Old" and "New" productions and performed separate analyses on each group. An observation is analyzed as an Old production if the associated rule is introduced in an earlier lesson. If the observation is made within the lesson in which the rule is introduced, it is a New production. So for example, conditionals are introduced in Lesson 3. The production *code-cond* is therefore New when used in Lesson 3, but Old when used in Lessons 4 - 12. Our regression model for Old productions is presented in Equation 1 below and our model for New productons in Equation 3. To make these more interpretable, we have derived equations from 1 and 3 in which the same independent variables, scaled in linear units, predict latencies in actual seconds. These are presented in Equations 2 and 4 for Old and New items respectively.

$$\text{log latency} = 1.31 - .01(L) - .25(\log O) - .26(\log P) \quad (1)$$

$$\text{latency} = 20.6\ (.97^{L})\ (O^{-.25})\ (P^{-.26}) \quad (2)$$

$$\text{log latency} = 1.36 - .03(L) - .30(\log O) - .15(\log P) \quad (3)$$

$$\text{latency} = 23.0\ (.94^{L})\ (O^{-.30})\ (P^{-.15}) \quad (4)$$

where L is lesson number, O is opportunity number and P is position.

Both regression equations account for highly significant proportions of the variance (r^2 = .17, $p < .001$, for Old productions; r^2 = .18, $p < .001$, for New productions) and all of the regression coefficients are significant ($p < .001$). The first thing to notice is that the intercept term, expressed in log seconds, is greater for New productions than for Old productions. On intuitive grounds, rules that are in the process of being learned should be used more slowly than well known rules. Overall, mean log latencies are in fact slower for New productions than Old (F [1, 840], = 8.87, $p < .001$). A second feature of these equations is that all the coefficients are negative. This means that as the values of our predictor variables increase, students' coding latencies decrease. It is interesting that both regression analyses identified the same independent variables. Among the predictors that we considered and rejected are the rated prior familiarity of productions, the number of goals pending in the problem for a particular observation, the depth of the symbol being coded within embedding function calls, and the position of the symbol on its line of pretty printed code. We will now consider each of the three independent variables that we have included in the models.

Lesson Effect: Learning to Learn.

Our students clearly improve over the course of the 12 lessons. The linear trends due to Lesson alone are negative and close in size to the Lesson coefficients in the three term models presented above (t[6409] = -7.29, $p < .001$, for Old productions; t[3350] = -12.73, $p < .001$, for New productions). Mean coding time in seconds is presented for each lesson in Figure 1. Although the curves show somewhat of an upturn in the final few lessons, their important feature is a general negative trend. The speedup due to lessons cannot be a simple practice effect. If it were, New productions would not show improvement: By definition, performance on these items is measured in only one lesson, namely the one in which they are first presented. Yet, the advantage of additional lessons is greater for New productions than Old items (t[9758] = 7.47, $p < .001$). A second possibility is that the Lesson effect is an "interface effect." The more experience students have with the tutor's display properties, for example, the more rapidly they can enter code. While the improvement on Old items may well be due to increased familiarity with the interface, we interpret the greater lesson effect for New items as evidence of yet another learning process.

Instead of an interface effect, we may be observing students' acquisition of schemas for coding LISP functions. For example, students may come to know that a LISP function must start with the function name followed (typically) by a fixed number of arguments. As the

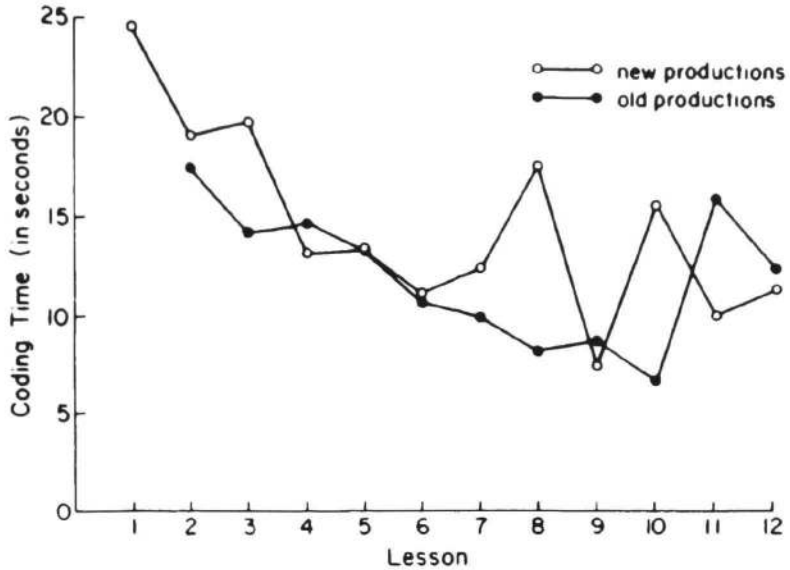


Figure 1: Mean coding time for Old and New productions as a function of lesson.

student encounters new functions, new productions can be created as instantiations of this general tendency. The benefit of assimilating new information with the help of schemas is well documented (*e.g.* Bransford, Barclay & Franks, 1972; Bower, Black & Turner, 1979). In the current context, we refer to this as the "learning to learn" phenomenon. One implication of the phenomenon is that learning certain LISP constructs will require students to update and elaborate their function schemas. If there is no longer a correspondence between the schema and the code students must write, then the benefits of the schema for learning New rules should be reduced until students revise their schemas. This should occur when the conditional structure and recursive structures are introduced. This material is presented in Lesson 3 (conditionals), Lesson 7 (numeric recursion), Lesson 8 (*cdr-recursion*) and Lesson 10 (*car-cdr* recursion). It is at these points in Figure 1 that we see the peaks for New productions, suggesting that the Lesson effect is modulated by lesson content and presumably the impact of content on student schemas.

Opportunity Effect: Knowledge Compilation.

The second term in both of our regression models is log opportunity to code a production within a lesson. An opportunity is essentially a trial in experimental terminology. It is therefore not surprising that students apply particular rules more quickly with practice. However, we have theortetical reasons for expecting that the improvment due to practice is not strictly linear. On the first opportunity to execute a New production, the student has only read the information that will form the basis of the rule. Within the framework of ACT* (Anderson, 1983), this *declarative* knowledge must be *proceduralized*, i.e. compiled into a production rule. Proceduralization explicitly builds certain long term information into the production that previously required maintenance in working memory. Because of the reduced demand on working memory, the student can apply the rule more efficiently. If this is so, we should see a sharp improvement in coding latencies early in the history of a production, followed by more gradual speedup. Once the new information is proceduralized, the resulting rule can be *tuned*, either facilitating or impeding its use, but the greatest change in a rule's performance is associated with its transformation from declarative to procedural knowledge

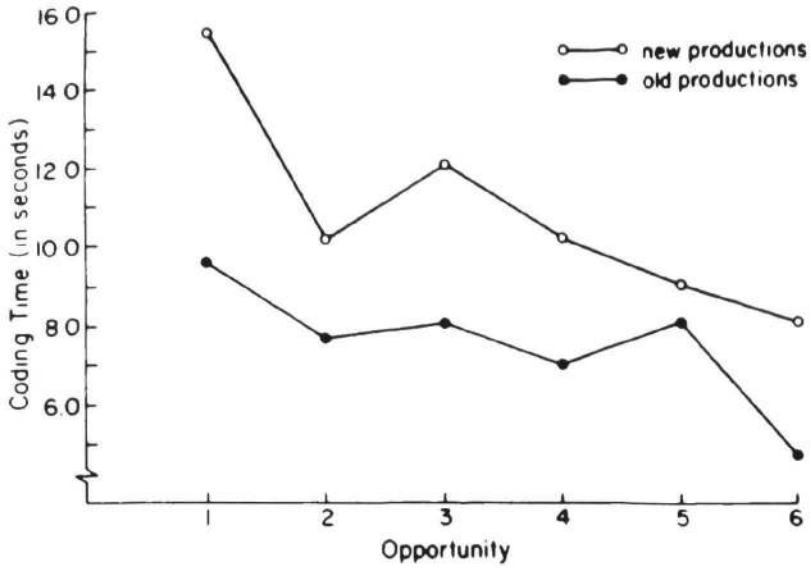


Figure 2: Mean coding time for Old and New productions as a function of opportunity.

This is largely the picture that emerges from our data. Students improve more on New productions than on Old ($t[9758] = 5.88$, $p < .001$) as you would expect if the New items reflect the transition from declarative to procedural kowledge. While we have performed our regression analyses on log coding latencies and log opportunity, the dramatic shifts in performance that we are looking for should be most evident in the untransformed data. In Figure 2, we plot coding time in seconds against actual opportunity count for Old and New productions. The most salient aspect of the Figure is the steep drop in coding time between the first and second opportunity. The improvement is on the order of 33% for New rules versus 20% for Old rules. It seems reasonable to us that this is the difference between proceduralization and tuning. It is also worth noting that at their slowest, Old productions fire about as fast as well practiced New productions. The suggestion is that there is a fundamental difference between the knowledge driving performance on the first opportunity for New productions and the remaining points on the graph. We would expect this if students are using declarative knowledge on this first observation alone.

Serial Position Effect: Plan Rehearsal.

Absolute serial position is the final performance factor that we will consider. This is simply the ordinal value of a production in the sequence of rules comprising a function definition. So for example, *defun* will typically be in the first serial position, *specify-function-name* in the second, and so on.[1] The negative regression coefficients for log absolute serial position tell us that the further into an exercise a student proceeds, the faster he or she will apply production rules. However, a closer examination of the data reveals systematic deviations from the linear trend captured by the regression models.

[1] Of course most rules can appear in a variety of positions.

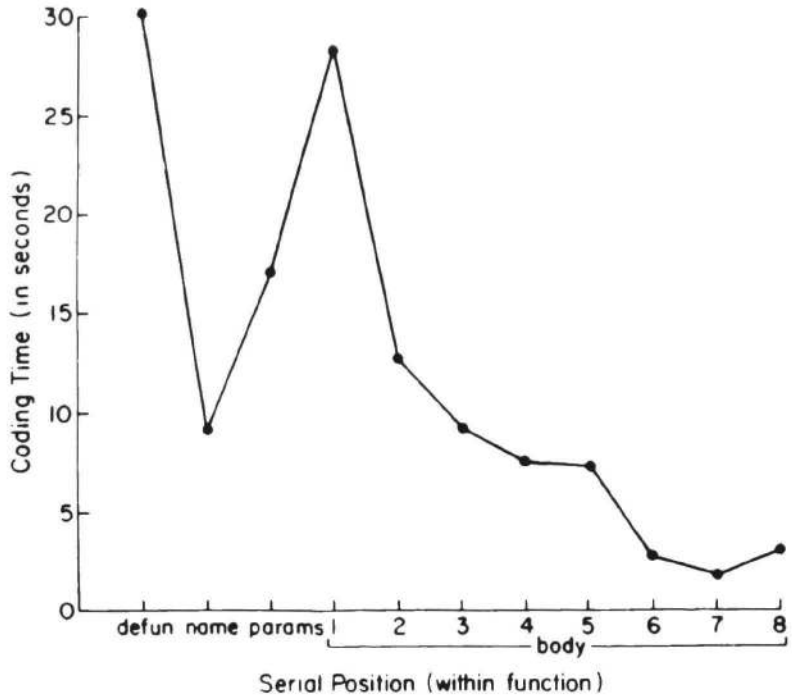


Figure 3: Mean coding time across serial positions in Lesson 2 functions.

We have assembled a profile of coding performance over serial position for the 12 exercises of Lesson 2. Both Old and New productions are included in this analysis which is presented in Figure 3. Latencies and serial position are untransformed in the plot to accentuate the non-linearity of the curve. We have distinguished the first three serial positions, which correspond to the same productions across problems, from the remaining positions, which are used to code the function body. What is striking about this curve is the twin peaks. The first peak likely reflects the final comprehension of the problem description and planning of the solution that precedes coding. The drop over the next two positions is not surprising since these productions are the only permissable actions at these points. We then see the second peak followed by a fairly systematic and negatively-accelerated speed up across serial positions as our regression analysis implies. We think this reflects an effect of formulating and rehearsing the plan. Repeated access and use of the plan should strengthen its encoding in working memory and so speed access to it. It should be noted that in our search for predictors log serial position proved to be a better predictor than other possible variables such as number of pending goals or depth of embedding of the code.

Conclusion

The LISP tutor has enabled us to analyze a complex skill in greater detail than is possible with more conventional methodologies. We feel there is a bright future for tutoring systems in psychological research. Despite the volumes of data and the fine grain size of our analyses, the picture we come away with is actually rather simple. Students' mastery of LISP is best captured by their progress through the curriculum (the lesson effect), the amount of

practice they receive on a particular production rule (the opportunity effect), and the location of that rule within the particular function that they are coding (the serial position effect). This simplicity is particularly striking.

We think that each of the three variables is in the main quite interpretable although they may also be picking up subtle complexities. The lesson effect seems to reflect the schema for working in LISP both with respect to the interface and the structure of LISP functions. The opportunity effect is a simple learning effect. It is noteworthy that the linear relationship between log latency and log opportunity indicates we have another instance of a power-law learning function (Anderson, 1982). The prediction is not as good if we use either linear time or linear practice. Finally, the Serial Position effect indicates the development of an initial plan which becomes rehearsed as it is used in writing the function. Each of these phenomena are hardly news in cognitive psychology. It is this fact that we regard as the news of our paper: Under appropriate decomposition (*i.e.*, as provided by the LISP tutor) acquisition of a complex skill is a matter of simple learning processes.

References

Anderson, J. R. (1982). Acquisition of cognitive skill. *Psychological Review, 89*, 369 - 406.

Anderson, J. R. (1983). *The architecture of cognition.* Cambridge, Ma.: Harvard University Press.

Anderson, J. R. & Reisser, B. J. (1985, April). The LISP tutor. *Byte*, pp. 159 - 175.

Bower, G. H., Black, J. B. & Turner, T J. (1979). Scripts in memory for text. *Cognitive Psychology, 11*, 177 - 220.

Bransford, J. D., Barclay, J. R. & Franks, J. J. (1972). Sentence memory: A constructive versus interpretive approach. *Cognitive Psychology, 3*, 193 - 209.

Ericsson, K. A. & Simon, H. H. (1984). *Protocol analysis: Verbal reports as data.* Cambridge, Ma.: The MIT Press.

Neisser, U. (1976). *Cognition and reality.* San Francisco: W. H. Freeman and Company

THE NATURE OF EXPERTISE IN THE CLINICAL INTERVIEW: INTERACTIVE MEDICAL PROBLEM SOLVING

David R. Kaufman and Vimla L. Patel
Centre for Medical Education
McGill University

The doctor patient interview is an interactive, goal-directed process; diagnosis being one of the primary goals. The clinical interview is perhaps the most important source of medical data that a physician has access to. Although diagnoses are most often made during the history-taking (Kassirer & Gorry, 1978), it is one of the least understood aspects of clinical cognition. In medical problem solving there have been comparatively few studies that have examined the acquisition and representation of clinical information. Elstein, Shulman and Sprafka (1978) used doctor-patient simulations to evaluate the performance of physicians. Barrows, et al., (1978) extended the research to evaluate the nature of clinical expertise along a developmental continuum, from novice medical student to expert physician. The principal goal of these studies was to characterize the general aspects of clinical performance. The emphasis was on process rather than content or knowledge. Their findings did not discriminate between between student and expert physician except for the accuracy of diagnostic hypotheses.

Recent investigations in medical problem solving have shifted from an emphasis on global aspects of clinical reasoning to a focus on the nature and content of medical knowledge used to solve a problem. Feltovich and colleagues investigated knowledge-based differences between expert and novice in the domain of pediatric cardiology (Feltovich et al., 1984). The results suggested that there are systematic differences in the structure of clinicians' knowledge-base, corresponding to certain levels of expertise that direct them towards particular inferences in generating diagnostic hypotheses. Patel and Groen (1986) demonstrated that expert physicians who accurately diagnosed a clinical case, developed explanations of the disease process that could be accounted for in terms of a forward-chaining strategy, moving from propositions in the stimulus text to conditions that suggested a component of the diagnosis. Expert physicians who misdiagnosed the case used a backward-chaining strategy, characterized by the posting of a potential cause and the subsequent generation of confirmatory evidence.

These studies provide converging evidence that it is primarily the differences in a clinician's knowledge base that allows them to effectively represent a problem and generate an appropriate solution. This would suggest that any characterization of expertise in the context of the clinical interview cannot be made independent of the knowledge necessary to solve the problem. The research presented here represents an attempt to investigate the clinical interview as a problem-solving situation. In particular, this study investigates expert-novice differences in the acquisition, representation and utilization of patient information in the course of the clinical interview.

THEORETICAL FRAMEWORK

It was determined that to effectively investigate this area we needed a coherent epistemological framework— identifying appropriate units of knowledge that correspond to medical problem solving and a characterization of the task environment and cognitive demands of the clinical interview The overall framework is described in greater detail in Kaufman (1987) and Patel, Evans, and Kaufman (in press).

The epistemological framework is adopted from a model proposed by Evans, Gadd and Pople (in press) who distinguish six levels at which clinical knowledge stratifies. Four of these levels are of interest to this research. **Observations** are units of information that are recognized as potentially relevant in the problem-solving context. They do not constitute clinically useful facts and many observations will be disregarded when formulating diagnostic hypotheses. **Findings** are comprised of observations that are anchored in the diagnostic context and have potential clinical significance. **Facets** are clusters of findings that are suggestive of prediagnostic interpretations. Facets reflect general pathological descriptions such as *aortic insufficiency* or categorical descriptions such as *endocrine problem.* **Diagnosis** is the level of classification that subsumes and explains all levels beneath it. The diagnostic label serves to summarize and organize the patient findings into a coherent and functional unit that facilitates therapeutic and management decisions.

Facets and diagnoses define the context and provide a framework in which observations and findings can be interpreted. This results in the instantiation of default expectations that guide the physician's subsequent questions. Observations form the minimal unit of the problem space. Observations

convey the propositional information that is subjected to inferential operations and results in the construction of higher order conceptual structures.

Medical problem solving has been described as an ill-structured task (Johnson, 1983). Ill-structured problems are ones in which the initial states, the goal state and the necessary constraints, are unknown at the beginning of the problem-solving process. In solving a patient's problem, the initial state, the goal state, and the elements of the problem space are undefined prior to the interview. The interim subgoal is to formulate a representation of the patient's presenting complaint. This as well as patient variables become the initial knowledge state. The clinician's representation of this initial state will in part determine subsequent inquiry. The permissible operators are essentially the range of questions asked by the clinician to elicit specific information. The application of these operators can be guided towards specific inquiry, such as evaluating a hypotheses or oriented towards general information gathering, such as assessing the patient's past medical history.

Data-gathering strategies can be guided by a goal to elaborate the general parameters of a problem-space; that is, to derive findings and meaningful relations among findings in the patient's history. This process is principally bottom-up, generating facets and differential diagnoses from the elements in a problem space. This is referred to as *reasoning diagnostically* (Patel, Evans & Chawla, 1987). Once the problem representation is sufficiently well elaborated, certain expectations are built up. The inquiry is directed towards eliciting specific findings that correspond to the interim interpretations (facet hypotheses) and diagnostic hypotheses that the physician has generated. This top-down process is referred to as *reasoning predictively*. It is expected that experienced physicians have a greater capacity to reason in the predictive mode, because they are more able to recognize patterns of elements in a problem that are associated with certain diagnoses.

EXPERIMENTAL METHODS

Fifteen volunteer subjects were selected: 5 endocrinologists (experts), 5 physicians in their third year of residency programs (intermediates), and 5 final year medical students (novices) who had completed their clinical training. Each subject was given 15 minutes to take the patient's history. The patient was a paid volunteer outpatient. Each session was videotaped and the dialogue was subsequently transcribed. The patient was a 22 year old oriental male who presented with two episodes of severe muscle weakness and other manifestations. This problem had been diagnosed as hypokalemic periodic paralysis associated with thyrotoxicosis, an uncommon disorder of the thyroid gland, involving episodes of paralysis associated with a marked fall in serum potassium. This problem can be decomposed into two components, *hyperthyroidism* and *hypokalemia*.

Diagnostic accuracy was evaluated according to whether the two diagnostic components, hyperthyroidism and hypokalemic periodic paralysis were present in the subjects' differential diagnoses. The transcribed dialogue was segmented into units of doctor-patient exchanges, which are physician-question/patient-response pairings. The method used to characterize the doctor-patient dialogue incorporated a pragmatic analysis of the discourse (described in Patel, Evans & Kaufman, in press), whose primary function is to characterize the information in focus during the course of the interview. This analysis is designed to evaluate how a clinician manipulates the context of the interview, to elicit observations, conclude findings, and derive meaning from higher order relations in the data. It also permits us to evaluate data-gathering strategies, as evidenced by sequences of questions elicited and findings concluded during the course of the interview.

Observations and **findings** are the main units of data used in this study to characterize the representation that the clinician is constructing. Observations in this analysis are units of information expressed by the patient or elicited by the physician in the course of the interview. The criterion for coding an observation is that it contributes unique information to the interview and is accepted by the physician.

Findings reflect a decision made by a physician concluding that a particular array of observational data contains a significant cue or cues that need to be accounted for in the diagnostic context. Findings are multi-faceted concepts that have certain well-formedness conditions. Certain propositional parameters such as location, duration, and sensation quality, need to be satisfied to conclude a specific finding. Findings are specific to particular individuals with unique presenting problems. However, it is possible to specify a relatively comprehensive set of generic findings. Towards this end, we appeal to a highly developed source of medical knowledge. Internist-I is an expert system for computer-assisted diagnosis in internal medicine (Miller, et al., 1984). It has an extensive knowledge-base incorporating over 550 diagnostic classifications and more than 4000 findings. Findings are determined to be concluded if they correspond in form to internist findings. This method is explicated in Patel, Evans, and Kaufman (in press).

Findings can be either negative or positive. A positive finding is a determination by the subject that a finding is characteristic of the patient's condition. A negative finding is a result of a conclusion that a particular finding is not indicative of the patient's problem. Negative findings are particularly useful in discriminating between competing diagnostic hypotheses. However, it is reasonable to speculate that the more the subjects elicit negative findings, the more likely the subject has not been able to develop an adequate representation of the problem.

A reference model was created, with the assistance of an expert endocrinologist, to identify and classify findings with respect to their importance in this particular case. The model identified the relevant findings in this case. Relevant findings are suggestive of this diagnosis and are contributory to the identification of the problem. In total, there were 16 findings designated as relevant.

The variables used in this study include the accuracy of diagnosis, the number of question/answer exchanges, the number of observations elicited, the number of total findings acquired, the number of positive and negative findings elicited and the percentage of relevant findings elicited during the interview. In addition, to these variables we can derive three measures of efficiency: a) the number of observations per question/answer exchange expressed as a percentage. This is designed to assess the efficacy of a subjects' ability to elicit potentially relevant information; b) the number of findings per question/answer exchange expressed as a percentage. This is a more specific index of a clinicians' ability to recognize significant information in the patients statements and to focus probes directly on the problem that lead to the generation of specific findings; c) the number of positive findings over question/answer exchanges as a percentage. This measure assesses the ability of a subject to develop a problem representation, anticipate candidate findings and reason predictively.

Three exploratory multivariate and two univariate analyses of variance were performed. The level of expertise was the independent variable for each analysis. The first multivariate analysis included question/answer exchanges, total number of observations and total number of findings as the dependent variables. In the second multivariate analysis, positive and negative findings were the dependent variables. In the third multivariate analysis, the first two efficiency measures, observations divided by exchanges and findings divided by exchanges were the dependent variables. The first univariate analysis used relevant findings as the dependent measure and the second univariate analyses used the third efficiency measure, positive findings over question/answer exchanges as the dependent measure.

RESULTS AND DISCUSSION

DIAGNOSTIC ACCURACY: Four out of five expert physicians accurately identified both components of the diagnosis. The other expert identified only the thyroid problem. One of the residents accurately identifed the problem, while two others identified the more general aspect, hyperthyroidism. One resident recognized the fact that the patient was suffering from periodic paralysis, but failed to identify hypokalemia as the causal agent. One subject did not recognize either component. No novice was able to completely diagnose the problem. However, three of the novice medical students recognized the thyroid aspect.

INFORMATION ACQUSITION: A summary of the results pertaining to question/answer exchanges, findings and observations are presented in Table 1. The first multivariate analysis with question/answer exchanges, observations, and findings as dependent variables yielded a significant group (level of expertise) effect ($F[2,12] = 4.64, p < .05$). There was also a significant group by measure interaction effect ($F[4,20] = 3.43, p < .05$). The endocrinologists had on average the shortest interviews as judged by the number of question/answer exchanges, followed by the residents with the medical students on average conducting the longest interviews. As the interaction effect would suggest, the pattern changes somewhat for the findings and observations. The residents elicited the greatest number of findings, followed by the medical students and the endocrinologists. In each of these 3 measures the endocrinologists generated significantly lower frequencies than either of the other 2 groups.

The second multivariate analysis, which compared the positive and negative findings generated, yielded no significant differences. It is assumed that positive findings have a bearing on evaluating or confirming working hypotheses, while negative findings are useful in discriminating between alternative hypotheses. In examining the group means, it becomes apparent that a pattern is emerging. As mentioned previously, the experts elicited fewer observations and generated fewer findings than the other groups, which on the surface would seem puzzling. However, the experts generated a comparable number of positive findings (14.4) to the residents (14.6), which was on average more than the students generated (13). This would suggest that the experts are more focused in that they need fewer questions to generate significant information and to conclude findings. Generally, the resident group generated almost as many negative

findings as positive findings. The comparably large number of negative findings are suggestive of the fact that this group had difficulty accounting for the myriad of problems the patient was presenting with and could not develop a coherent problem representation. The generation of numerous negative findings would also indicate that these subjects were considering multiple hypotheses and were attempting to rule-out as many as possible. The medical students, on average, elicited somewhat fewer negative findings than did the residents. This may suggest that those subjects who were able to identify the thyroid aspect were content that they had correctly diagnosed the problem. The residents recognized that there were many discrepant findings that could not be accounted for by hyperthyroidism and pursued alternative explanations. The experts develop a coherent representation of the distinctive pattern of findings and elicit fewer negative findings because they do not perceive as great a need to rule out competing diagnoses.

TABLE 1: Categories of information acquisition by level of expertise.

	EXPERTS		RESIDENTS		NOVICES	
CATEGORY	MEAN	S.D.	MEAN	S.D.	MEAN	S.D.
Question/Answer Exchanges	77.4	31.9	119.4	19.1	122.8	20.1
Total Observations	41.6	19.2	61	7.3	56.8	9.9
Total Findings	20.8	2.9	27.4	4.5	23	4.8
Positive Findings	14.4	2.5	14.6	3.1	13	3.8
Negative Findings	6.4	3.6	12.8	5.6	10	4.9
Percentage of Relevant Findings	72	13	60	19.5	45	8.2

The efficiency measures are supportive of the findings suggesting a developmental continuum in the selectivity of information. The multivariate analysis with the first two efficiency measures, observations divided by exchanges and findings divided by exchanges, as dependent variables, resulted in significant group differences ($F[2,12] = 8.69$, $p < .01$). The expert endocrinologists had on average the highest efficiency scores on both measures followed by the residents and students. The differences are most salient on the second measure, the percentage of findings generated per question/answer exchange. This indicates that the experts are more efficient in identifying cues in the patient data and ask the appropriate questions which lead to the generation of findings. The medical students are the least efficient in focusing probes on aspects of the problem that lead to the generation of findings. Since they do not have much clinical experience, they may not be as capable of recognizing the specific contexts where findings are manifested. The univariate analysis of variance on the third measure of positive findings per question/answer exchange also resulted in highly significant group differences ($F[2,12] = 7.28$, $p < .01$). The expert group had a significantly higher group mean (18.6) on this measure than either the residents (12.2) or the students (10.6). This would suggest that the expert group tend to recognize aspects of the problem early on and attempt to pursue hypotheses that suggest candidate findings that lead to a component of the diagnostic solution. While the other two groups have greater difficulty in isolating the problem and are less capable of anticipating potential findings.

The differences that have emerged thus far are suggestive of differences related to the selective acquisition and processing of information. The results pertaining to the percentage of relevant findings generated are in the expected direction. The univariate analysis comparing the relevant findings elicited yielded a significant group effect ($F[2,12] = 4.46$, $p < .05$). The endocrinologists generated 72% of the available relevant findings, followed by the residents with 60% and the medical students who were only able to elicit 45% of the relevant findings.

To accurately diagnose the case, it is essential to elucidate the most significant findings in the case. Perhaps the most telling index of the selective acquisition is the percentage of relevant findings over the total number of findings generated. This reflects an ability to isolate and elaborate upon the most fundamental aspects of the clinical problem. It is this statistic that reveals the most striking differences

between groups. Fifty percent of all findings elicited by the expert physicians resulted in the generation of a relevant finding. This was considerably more than the other two groups, with the residents yielding 33% and the medical students yielding 31% of the total findings they generated as relevant. This indicates that the endocrinologists are able to recognize patterns in the data and focus on the pertinent information leading to components of the diagnosis. The residents generally can identify much of the relevant information. However, they do not possess the same domain specific knowledge and are consequently less selective in the information they elicit. The students do not have the knowledge nor the clinical experience to generate the relevant information that has to be accounted for.

PROBLEM REPRESENTATION STRATEGIES: Thus far, it is at the level of findings that the most salient differences have emerged. This section attempts to elucidate the broader spectrum of context and describe the nature and sequence of information in focus. This accounting provides a framework to discuss differences in the hierarchical structuring of information and the use of strategies in the construction and modification of a patient problem representation. This analysis is primarily descriptive with illustrations from individual protocols provided to highlight certain points. In the context of the clinical interview, the problem space is not defined a priori, it is constructed in the course of the evolving interaction. We can assume that there are multiple pathways to the goal state of diagnosis, which vary greatly in their efficiency and efficacy. It is proposed that through examining the sequence and nature of information in focus, we can characterize the strategic construction of a problem representation.

Predictive reasoning involves the generation of an initial problem formulation and the subsequent pursuit of specific candidate findings, driven by a hypotheses or small subset of related hypotheses. It is analogous to a forward-chain of reasoning. It is a very powerful strategy with an associated degree of risk. It has the effect of controlling uncertainty, by delimiting the number of variables that must be accounted for in the problem resolution process (Patel, Evans & Chawla, 1987). It is an efficient strategy that minimizes cognitive load and maximizes the allocation of resources towards the assessment of a particular hypotheses or a small subset of related hypotheses. The associated risk is that the scope of probing for information will be too narrow and essential elements may be overlooked.

An excerpt from the interview of expert 2 is presented in Table 2. At this point in the interview this subject had documented the patient's presenting complaint. In question/answer exchange 20, she focuses on one of the cardinal signs of hyperthyroidism, *heat intolerance*. She provides an exemplar, describing a circumstance in which the patient may be most cognizant of this problem. The patient at this point rejects that characterization. She continues to pursue this line of reasoning, eliciting several of the critical findings characteristic of hyperthyroidism. After establishing that the patient had been eating more (exchange 24), she attempts to link the thyroid component with the episodes of muscle paralysis. In exchanges 25 and 26, she focuses on the particular foods that the patient had been indulging in. Carbohydrates predispose patients to attacks of hypokalemic paralysis. After the patient agrees with this description, the physician subsequently attempts to determine whether the patient recalls eating a carbohydrate meal prior to the second muscle attack. This protocol documents the efficacy of predictive reasoning in generating findings from facet-level hypotheses and in integrating facets in the formulation of diagnostic hypotheses.

TABLE 2: DOCTOR-PATIENT DIALOGUE EXPERT 2

20. D. No. Had you noticed that you were minding the heat any more than other people you live with, or other people in the room?
P. Umm, no
21. D. No. Did you notice any tremor of your hands?
P. Well, yeah.
22. D. How long before?
P. Four months.
23. D. Four months. Did you notice any weight loss?
P. Around ten pounds.
24. D. Ten pounds. Were you eating more?
P. Yes.
25. D. Yes. What kinds of things were you eating?
P. Uhh, meat; beef, well, fish, sometimes ribs.
27. D. Well, sweet things, or macaroni, noodles, kind of carbohydrate things?
P. Well, yeah.
30. D. Do you recall if you were eating anything unusual just before this attack happened?

P. Umm, No.

An excerpt from a residents (R3) is presented in Table 3. In this situation, the interview had just begun and the subject was attempting to characterize the patient's presenting complaint. In the excerpt the topic in focus is leg muscle weakness. The resident is attempting to characterize the problem in terms of a circumstance that he could recognize as being exemplary of a medical context he could explain. Despite the fact that the interview had just started, the subject makes no attempt to elaborate on the patient's complaint. He presents the patient with a series of probes in the form of binary response choices (e.g. yes or no), using exemplars depicting circumstances in which the weakness may manifest itself.

In the course of this exchange, he fails to pick up on certain critical pieces of information. In question/answer exchange 6, for instance, the subject focuses on the periodic pattern of the weakness. The patient responds that it has only happened twice, a critical piece of information for determining that this is *periodic* as opposed to *generalized weakness*. The clinician continues to attempt to characterize the nature of the weakness, under the assumption that this is a problem of *chronic muscle weakness*. He provides the patient with potential situations in which the weakness may manifest itself, demonstrating that he has a causal model of different leg muscle problems and is attempting to instantiate the corresponding findings. In the course of pursuing these hypotheses, he has overlooked three essential findings, including the aforementioned finding. In exchange 9, the patient informs the physician that it just happened suddenly. From his subsequent probes, it appears that he does not conclude *weakness of sudden onset*, but rather infers that the weakness is generalized, but intermittent in severity. The follow-up questions, only a sample of which are included here, focus on potential palliative and exacerbating factors that impact on the weakness.

In this stretch of discourse, it is apparent that he has misrepresented the patient's complaint. The initial formulation of a problem representation constrains the search space and determines the probes that follow it. This was clearly the case in this protocol. This subject was unable to characterize any aspect of the patient problem, primarily for the reason that he was working under the assumption that this was a problem of *chronic muscle weakness*, and never triggered the appropriate facet of *acute periodic weakness*. As a result, the subject introduced greater uncertainty into the problem space, by generating multiple competing facet-level hypotheses, none of which adequately characterized the patient's condition. This is apparent throughout the protocol and even in the diagnoses listed in his differential.

TABLE 3: DOCTOR-PATIENT DIALOGUE RESIDENT 3

3. D. What's the problem?
P. I have muscle problems.
4. D. You have muscle problems? In your legs? What do you have, muscle weakness, or?
P. Weakness, no pain.
5. D. No pain at all. How long have you had the weakness?
P. Three months.
6. D. Three months. Is it there all the time, or does it come and go?
P. It's coming and going. Its happened twice.
7. D. Ok. Do you find let's ..say...Do you have the weakness when you wake up in the morning?
P. No.
8. D. Do you have to do a certain amount of exercise before you get weak?
P. Ah, no.
9. D. So, let's say...let's say you could walk down the street and as soon as you started walking you could notice that there is a weakness. Would you say that you walk a distance first and then you get weak?
P. Ahh... not really, it just suddenly felt weak.
10. D. And do you have periods let's say where you feel weak and then you feel normal again and then you are weak? Like that?
P. Yeah.
12. D. Ok. Is there anything you know that makes it better? You feel less weak ...or.
P. No.

The primary differences between the experts and the residents protocols are reflected in the ability to formulate an initial representation of the patient problem. The expert effectively characterizes the patient's presenting complaint and subsequently pursues a high-yield knowledge-based strategy to confirm a diagnostic hypothesis. The resident fails to elaborate the patient's presenting complaint and as a consequence, pursues general data-gathering strategies that continuously widen the scope of diagnostic

candidates without any forward movement towards the solution state. The differences can be attributed to domain-specific knowledge and domain-specific experience with patient's presenting with similar histories.

As expected, there were differences in terms of the accuracy of diagnosis, with the experts generating the most accurate diagnoses, followed by the residents and medical students, respectively. The most global measures of subjects' performance such as the total number of observations and findings generated are neither adequate indices of expertise nor meaningful indicators of successful performance. Novices and residents tended to elicit more total observations and findings than the experts did. It appears that the subexpert groups needed to elicit more observations to generate findings and more findings to derive facets. However, it is the nature of the findings and not the number of findings that is associated with successful performance. Findings need to be coordinated and organized under the relevant facets, which in turn are extrapolated to the appropriate diagnostic context. There were several instances where elements of a significant finding were elicited but because of inappropriate background assumptions were misconstrued.

The study demonstrated that experts have a greater capacity to elicit the appropriate information in context and elaborate the constraints of a problem-representation. Given a complex problem, intermediate subjects experience difficulty in integrating information from multiple sources. They tend to pursue hypothesis-driven strategies that lead them down multiple search paths. Novice medical students experience greater difficulty at the local level–deriving findings from the patient's observations.

In general, the results of this study suggest that the selective acquisition and efficient utilization of information are hallmarks of expertise. This finding is consonant with results of other problem-solving studies in medicine and in other domains that indicate that the rapid, systematic application of knowledge to problems and the use of effective knowledge-based strategies leading to accurate solutions distinguish performance at different levels of expertise and individuals varying in their competency.

REFERENCES

Barrows, H.S., Feightner, J. W., Neufeld, V. R. & Norman G. R. (1978) Analysis of the Clinical Methods of Medical Students and Physicians. Final Report, Ontario Department of Health, Hamilton, Ontario.

Elstein, A. S., Shulman, L. S., & Sprafka, S. A. (1978). Medical problem solving: An analysis of clinical reasoning. Cambridge, MA: Harvard University Press.

Evans, D. A., Gadd, C. S., & Pople, H. E. (in press). Managing coherence and context in medical problem-solving discourse. In Evans, D. A. & Patel, V. L.(Eds.) Cognitive Science in Medicine. Boston: MIT Press.

Feltovich, P. J., Johnson, P. E., Moller, J. H., & Swanson, D. B. (1984). LCS: The role and development of medical knowledge in diagnostic expertise. In W. J. Clancey & E. H., Shortliffe (Eds.), Readings in medical artificial intelligence. Reading, MA.: Addison-Wesley.

Johnson, P. E. (1983). What kind of expert should a system be? The Journal of Medicine and Philosophy, 8, 77-97.

Kaufman, D. R., (1987). Representation and utilization of information during the clinical interview in medicine. Unpublished M.A. Thesis, McGill University, Montreal, Quebec.

Miller, R. A., Pople, H. E. & Myers, J. D. (1984). Internist-I, an experimental computer-based diagnostic for general internal medicine. In W. J. Clancey & E. H., Shortliffe (Eds.), Readings in medical artificial intelligence. Reading, MA.: Addison-Wesley.

Patel, V. L., Evans, D. A., & Chawla, A. S. (1987). Predictive versus diagnostic reasoning in the application of biomedical knowledge. In Proceedings of the Ninth Annual Meeting of the Cognitive Science Society, Seattle, WA.

Patel, V. L., Evans, D. A., & Kaufman, D. R. (in press). Cognitive framework for doctor-patient communication. In Evans, D. A. & Patel, V. L.(Eds.) Cognitive Science in Medicine. Boston: MIT Press.

Patel, V. L., & Groen, G. J. (1986). Knowledge based solution strategies in medical reasoning. Cognitive Science, 10, 91-116.

ACKNOWLEDGEMENTS: I would like to thank all the volunteers who participated in the study. This research was supported in part by a Natural Science and Engineering Council of Canada grant (# 82598) to Vimla Patel. I wish to express my appreciation to Dr. Y. Patel who assisted in the medical aspects of the study and Anoop Chawla, Stephen Chase, José Arocha,and Aldo Braccio for their editorial assistance.

A Dynamical Theory of the Power-Law of Learning in Problem-Solving

Jeff Shrager
Tad Hogg
Bernardo A. Huberman

Xerox PARC
Palo Alto, California

March 24, 1988

Abstract

The ubiquitous power-law of practice has been a touchstone of cognitive models. It predicts that the speed of performance of a task will improve as the power of the number of times that the task is performed. In this paper we derive the power-law from a graph dynamical theory of learning by considering changes in problem-space graph topology due to the addition of operators, and alterations in the decision-procedure used to decide which operator to apply at a particular step. The general approach of applying dynamical principles to cognitive problems holds much promise in unifying other areas of learning and intelligent activity.

Keywords: Learning, Chunking, Macro Operator Formation, Power-Law Speedup, Problem-Solving, Practice.

1 Introduction

The power-law of practice (i.e., Crossman, 1956, and many others) predicts that the speed of performance of a task will improve as the power of the number of times that the task is performed. This appears to be a ubiquitous property of human learning, and has been a touchstone of cognitive models such as SOAR (e.g., Laird, Rosenbloom, & Newell, 1986) and ACT* (Anderson, 1983). In this paper we derive the power-law of learning from a graph dynamical theory (Huberman & Hogg, 1987). This law results from changes in problem-space graph topology due to the addition of operators, and alterations in the decision-procedure used to decide which operator to apply at a particular step.

We begin by describing a simple "Bit Game" which will be used in our experiments. We then sketch the derivation of the power-law for operator addition and decision-procedure improvement. Next, we verify our approach by computer simulations of the Bit Game which isolate the relevant features of the theory. Finally, we discuss the importance of this approach to modeling learning, and briefly contrast it with existing accounts of power-law improvement.

2 The Bit Game

Many tasks can be viewed as the search for a path through a problem-space graph, where nodes represent states of the problem and links represent operators that move between states (Newell &

Simon, 1972). In order to illustrate the processes of learning, and verify the theory, we consider a simple problem called the "Bit Game" which is analogous to many well-defined problems, but has a simple problem-space. A problem state in the Bit Game is a B-bit binary vector (as: 01010). We will generally use a five-bit vector in these examples. A "trial" begins from an arbitrary initial state, say: 00000, and the goal is to change that state to some other arbitrary vector, say: 11111.

Each operator is composed of from 1 to B elements indicating a particular bit in the vector which should be flipped if it matches in the current state. Operators only specify the bits in the state which actually change. Operators can be written as "pattern → result" pairs, with question-marks ("?") where the operator pattern says nothing about a particular bit position. The operator "?1?1? → ?0?0?" will take the state: 11010 to 10000, or the state: 11111 to 10101, but will not apply to the state: 00000 because the indicated bits are not ones.

Operators vary in their generality. The one-element operators, such as: "0???? → 1????", apply to 16 states: (00000, 00001, 00010, ... 01111), and so connect together 32 nodes in the space of all states. We begin playing the Bit Game with all the ($2B$) one-bit operators (10, in the case of a 5-bit game). This set forms the B-dimensional hypercube and completely connects the space.

The number of links traversed on a trial (i.e., the number of steps required to find the goal state from the initial state) measures performance. A series of trials, beginning with a common initial problem-space and with learning between each trial, will be called a problem-solving "run". When there are several applicable operators per step, a decision-procedure is used to choose one. There are many reasonable ways to decide among different operators. For certain organizations of operators and choices of initial and goal states, some of these decision-procedures are more effective than others.

2.1 Adding Operators

After a trial is completed, learning may take place in one of the two ways: either by changing the problem-space by adding operators (links), or by changing the decision-procedure. For example, SOAR (Laird, et al., 1986) adds new operators which summarize the results of sub-problem-solving. In the Bit Game we add the operator that most generally summarizes the solution of the game just played. That is, suppose we begin with this game: 01110 =>10101, and find a solution. We can gain the effect of summarizing the "subproblem-solving" for that game by forming the operator that will solve this game in one step. That is, for this case we add the operator: "01?10 → 10?01" because it is the most general operator that will solve this game in one step. Notice that the "?" element of this operator appears because the third bit did not change between the initial and goal state in this trial, and so this new operator connects together two pairs of states in the problem-space.

2.2 Improving the Decision-Procedure

Decision-procedures can be arbitrarily complicated algorithms, and can be changed in complex and arbitrary ways, leading to entirely different problem-solving behavior. Generally, however, a decision-procedure is only radically changed in the face of some new insight into the problem structure. Without such an extreme change it only makes sense to either slowly vary the parameters controlling the decision-procedure in order to try to hill-climb into a best-solution mode, or to vary them randomly, hoping to discover a good decision-procedure serendipitiously. We consider changing between a very bad ("poor") decision procedure, and a very good ("optimal") one.

The "optimal" decision-procedure finds the fastest way to the goal. For each operator that can apply in the current state one can ask how many of the goal bits will be correctly set if that rule

were applied. We then randomly choose an operator from among the ones that score highest on this measure.

The "poor" decision-procedure is a random walk with a 1-ply lookahead: We search at random until we are one-step away from the goal, and then take that one step directly to the goal. Since as operators are added to the problem-space it becomes more connected, one has to do less random search in order to run into a path leading to the goal, but it is also easier to get off the path.

3 Analysis

In this section we derive the behavior to be expected from the addition of new operators and improvements in local decision-procedure. A problem space consists of a connected graph, G, with n nodes representing various problem states, and links representing instances of possible operators. We now show that learning due to the addition of links in the graph, or improvements in the decision-procedure, lead to a gradual reduction in path lengths with a corresponding gradual improvement in performance which is an power-law of the number of trials.

We are interested in learning behavior for situations involving a large number of states and typical problem spaces rather than any specific one. This can be elucidated by assuming that the initial set of operators are distributed at random, and that new links are added independently of one another. Alternatively, instead of a specific number of links, we can assume that the links between each pair of states each exist with independent probability p. Although such a probabilistic description appears to differ from the case of a fixed number of operators, the resulting properties are known to be the same when large graphs are involved (Bollobas, 1985). Moreover, this approach simplifies the mathematical derivations. As new operators are learned during the trials, p will correspondingly increase. Notice that p must be greater than $(\ln n)/n$ for the graph to be completely connected.

Similarly, to explore the range of decision-procedures that lie between the optimal decision-procedure and the random walk we use a simple descriptive model of the effectiveness of the decision-procedure in which, at any node during the search for the goal, each unproductive link is eliminated with probability $1 - \rho$. Improvements in the decision-procedure corresponds to a decrease in ρ and change the problem from an exponential random walk to a linear drift toward the goal (Huberman & Hogg, 1987). This probabilistic model describes the behavior of the algorithm when applied to many decisions, but does not necessarily require that choices be made randomly. Note that $\rho = 0$ corresponds to a perfect decision-procedure in which search and backtracking are never required, while $\rho = 1$ corresponds to a random walk on the graph. We assume that $p(T)$ and $\rho(T)$ are given functions of the number of completed trials T and investigate the consequences of their changes. That is, we want to obtain an expression, $s(T)$, relating the expected number of steps, s, required to obtain a solution to the number of trials (T).

To obtain an explicit expression for s in terms of p and ρ, we consider a simplified model which incorporates the essential features of problem-solving. First, we assume that all nodes of the graph have the average number of links $\mu = (n - 1)p$. Thus, we are left with a regular graph of n nodes and uniform branching ratio μ. Second, we assume that the cycles in the graph are long so that, in general, at any node there will be one link (one step) to a node closer to the goal while the others are one step farther away. In this limit the behavior will be similar to a walk on a tree. Because of the initially exponential growth in the number of nodes with distance, the initial and goal nodes will typically be separated by the diameter of the graph, which can be approximated as $D = \ln n / \ln \mu$, the maximum distance between points in the tree corresponding to the graph. From most nodes, there is only one choice which gets closer to the goal state and $\mu - 1$ choices which move farther from the goal.

However, the decision procedure eliminates each incorrect choice with probability $1 - \rho$ so there are (on average) effectively only $(\mu - 1)\rho$ incorrect choices at each node. Thus s will be approximated by the average number of steps required to reach the goal using this process.

This analysis gives the expected behavior of s as a function of p (i.e., the topology) and ρ (i.e., the decision effectiveness). For instance, when the decision-procedure is weak (ρ near 1), steps toward the goal will be relatively rare (recall that μ is at least as large as $\ln n$). Thus the system's behavior will be exponential. When $\mu\rho$ is much larger than one, one obtains:

$$s \approx 2(\mu\rho)^{\ln n / \ln \mu} = 2n\rho^{\ln n / \ln \mu}$$

In this case, when $\rho = 1$ (so that the decision-procedure is no better than making random choices), increasing the number of links has no effect on the time to solve the problem. More generally (i.e., $\rho < 1$), increasing the number of links will result in a gradual increase in s, the expected number of steps required to solve the problem, because the smaller diameter of the graph is more than balanced by the increased difficulty of choosing the correct operator from among the larger number of choices. Notice, however, that when connectivity becomes large, many paths will become equally good, resulting in an effective decrease in ρ. Recalling that μ can range from $\ln n$ to $n - 1$, we illustrate this behavior for $\rho = 0.9$ and $n = 100$ in Figure 1 (a).

To obtain the benefit of added links, the decision-procedure must improve as new links are added. For example, if it improves fast enough to keep $\mu\rho$ constant as links are added, the corresponding decay is illustrated for $n = 100$ and $\mu\rho = 5$ in Figure 1 (b)

Conversely, when the decision-procedure is strong (ρ near 0), the system's behavior is given by:

$$s \approx (\ln n)/(\ln \mu)$$

Thus, as long as the decision-procedure improves as fast as new links are added, one obtains a power-law decay in $\ln \mu$ as shown in the case of $n = 100$ in Figure 1 (c).

4 Experimental Verification

In order to validate our approach in actual problem-solving, we simulated operator addition and decision-procedure improvement in the Bit Game.

In our simulations, we always randomly choose a start and goal state, solve the problem according to some decision-procedure, and record the performance. The possible games are uniformly distributed among the 2^{2B} bit configurations.

Figure 2 plots the average (over 100 runs) search path length over 500 trials for the 7-bit Bit Game. In this instance we used the optimal decision-procedure. The path length decreases according to a power-law, as predicted, confirming that we have achieved at least first-order analogy to Rosenbloom's (1986) model in which the power-law is accounted for in terms of chunking of subgoal hierarchies. However, this analysis is more general than Rosenbloom's theory in the sense that we incorporate both improvements due to adding operators and due to improvements in the decision-procedure. Furthermore we predict a power-law for any sort of link addition, whereas Rosenbloom predicted power-laws only in the case of subgoal chunking. Furthermore, our theory is able to predict the precise exponents (if we know the properties of the particular problem-space graph and decision-procedure) whereas normally there are two free parameters in psychological power-law models.

In order to simulate improving the decision-procedure we explicitly incorporated the descriptive parameters of our model. Specifically, we found all applicable operators from the current state and then selected one from the set according to a function of the number of trials: $\rho(\tau)$. Thus, ρ begins at

1.0 and moves to 0.0 linearly throughout the run. In order to do this we first order the operators as for the optimal decision-procedure. We then separated these into the best ones (i.e., ones with the same highest goodness rating) and all the rest (i.e., ones that do not have that particular highest rating). Next we deleted each of the non-optimal operators from the set with probability $1 - \rho$. Finally, we chose one operator at random from the union of the remainder of the non-optimal set, and all the optimal operators. When $\rho = 0.0$ all of the non-optimal operators will be deleted, leaving only the optimal ones. This results in optimal problem-solving. When $\rho = 1.0$, all of the non-optimal operators are left in the set, making the decision-procedure into a random walk.

Figure 3 shows the results of the 7-bit Bit Game, with the above modifications, averaged over 365 runs and tracked through 200 trials. ρ decreases linearly along the independent axis in steps of 0.05. Again we observe the predicted power-law decrease in path length as the decision-procedure improves.

5 Discussion

Shepard (1987) raised the call for a law-like science of psychology, with the power of the laws of physics. In this paper we have taken a step toward Shepard's goal by presenting the outline of a unified theory of problem-solving learning phenomena based upon the mathematics of graph dynamics. Although some researchers have approached a unified model of phenomena and mechanisms (e.g., Anderson, 1983; Rosenbloom & Newell, 1981) these theories contain ad hoc or overly specific processes and assumptions. Anderson, for instance, obtains the power-law by a rule strengthening mechanism which operates according to a power-law. Rosenbloom's account of the source of the power-law is restricted to addition of operators resulting from the chunking of problem-solving subgoals. Thinking of a problem-space as a graph is a commonplace interpretation of problem-solving (dating from Newell & Simon, 1972), but to our knowledge, no one has tried to understand the relationship between graph dynamics and problem-solving learning phenomena.

The psychological mechanisms involved in learning by doing can include method selection (Crossman, 1959), method optimization (Cheng, 1985), and changes in the mechanisms that choose which operator to apply in a given situation (the decision-procedure). All sorts of changes in method and operators can be modeled as changes in the topology of the problem-space graph due to either restructuring (e.g., Shrager, 1987) or the construction of new operators (Korf, 1985; Rosenbloom, 1986). In many tasks, the synergy of mechanisms results in the well documented power-law of practice (Anderson, 1983; Fitts, 1964; Rosenbloom & Newell, 1981) in which the speed of performance increases rapidly at first, and then more slowly as performance becomes highly skilled.

We have shown that by applying the theory of graph-dynamics to the view of a problem-space as a graph, we can capture, explain, and experimentally demonstrate the power-law in a general way without recourse to explicit power functions. The power-law was found in two different cases. First, in the case of adding new operators to the problem space. This is a more general result than the results of Rosenbloom (1986) in that he studied one kind of new operator – the memory of the results of subproblem-solving. Second, we predicted and found a power-law in the improvements of decision procedures.

Several additional points are worth mentioning. First, merely adding links to a problem-space graph without an already relatively good decision-procedure will simply make things slower because there will be more ways of getting lost. This is the prediction seen in Figure 1 (a), and accords with another well-known psychological result: interference (e.g., Smith, Adams, and Schorr, 1978). Presumably either one starts out with a moderately good decision-procedure, or else as learning takes place, the person is both improving his decision-procedure in addition to learning new problem-space

links. The combination of effects – i.e., adding together two power-laws of different exponent, intercept, and out of phase with one another, can lead to complex phenomena ranging from power-laws, to steep jumps (up or down) superimposed on smooth performance, and, interestingly, to the precise behavior generally accorded to the paradox of the expert (Smith, Adams, and Schorr, 1978). This is the phenomenon in which the learner becomes worse until he knows a lot about the domain, and then performance improves dramatically. This suggests looking to decision-procedure improvement as a possible account for this paradox. Finally, although not discussed here, we have also uncovered another sort of phenomenon: i.e., sudden performance improvements in certain kinds of concept acquisition tasks (such as the tasks analyzed in Bourne & Restle, 1959) (these results are in preparation).

In sum, we feel that the general approach of applying dynamical principles (Huberman & Hogg, 1987, Shrager, Hogg, & Huberman, 1987) to cognitive problems holds much promise in unifying other areas of learning and intelligent activity.

Acknowledgments

We particularly thank Jeff Kephart for discussion of these issues and comments on the paper. Stu Card, John Lamping, and Scott Stornetta contributed to our thoughts on this research.

References

Anderson, J. R. (1983). The Architecture of Cognition. Harvard University Press, Cambridge, Mass.

Anzai, Y. & Simon, H. A. (1979). The Theory of Learning by Doing. Psychological Review, 86(2), 124-140.

Bollobas, B. (1985). Random Graphs. Academic Press, New York.

Bourne, L. E. & Restle, F. (1959). Mathematical Theory of Concept Identification. Psychological Review, 66(5), 278-296.

Cheng, P. W. (1985). Restructuring Versus Automaticity: Alternative Accounts of Skill Acquisition. Psychological Review, 92(4), 414-423.

Collins, A. M. & Loftus, E. F. (1975). A Spreading-Activation Theory of Semantic Processing. Psychological review, 82(6), 407-428.

Crossman, E.R.F.W. (1959). A Theory of the Acquisition of Speed-Skill. Ergonomics, 2, 153-166.

Huberman, B. A. & Hogg, T. (1987). Phase Transitions in Artificial Intelligence Systems. Artificial Intelligence, 33(2), 155-171.

Hogg, T. & Huberman, B. A., (1987). Artificial Intelligence and Large-scale Computation: A Physics Perspective. Physics Reports, 156(5), 229-310.

Korf, R. E. (1985). Macro-Operators, a Weak Method for Learning. Artificial Intelligence, 26(1), 35-77.

Laird, J., Rosenbloom, P., & Newell, A. N. (1986). Chunking in SOAR: The Anatomy of a General Learning Mechanism. Machine Learning, 1(1), 11-46.

Newell, A. N. & Rosenbloom, P. (1981). Mechanisms of Skill Acquisition and the Power Law of Practice. In J. R. Anderson (ed.), Cognitive Skills and Their Acquisition. Lawrence Earlbaum Associates, Hillsdale, N.J.

Newell, A. & Simon, H. A. (1972). Human Problem Solving. Prentice Hall, New Jersey.

Rosenbloom, P. (1986). The Chunking of Goal Hierarchies. in Laird, J., Rosenbloom, P. & Newell, A. N. Universal Subgoaling and Chunking. Kluwer Academic Publishers, Boston.

Shepard, R. N. (1987). Toward a Universal Law of Generalization for Psychological Science. Science, 237, 1317-1323.

Shrager, J., Hogg, T. & Huberman, B. A. (1987). Observation of Phase Transitions in Spreading Activation Networks. Science, 236, 1092-1094.

Shrager, J. (1987). Theory Change via View Application in Instructionless Learning. Machine Learning, 2(3), 247-276.

Smith, E. E., Adams, N. & Schorr, D. (1978). Fact Retrieval and the Paradox of Interference. Cognitive Psychology, 10, 438-464.

Figure 1: Analytic Predictions.

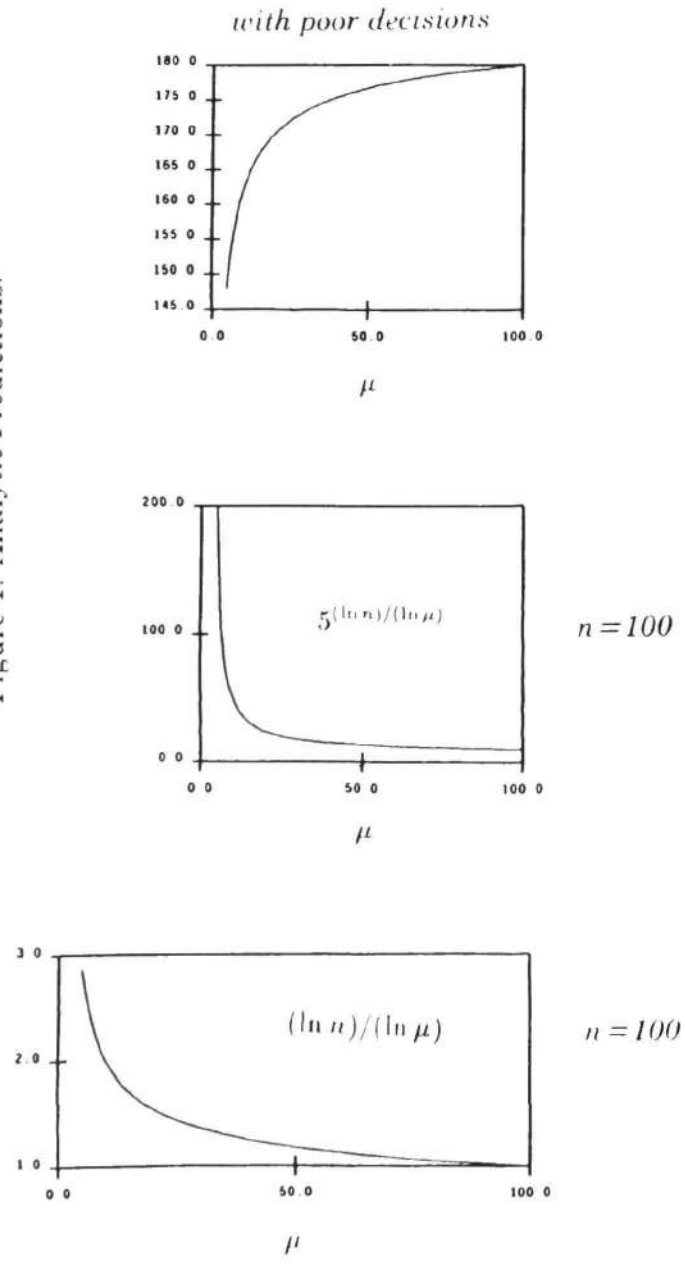


Figure 2: Bit Game power-law due to chunking.

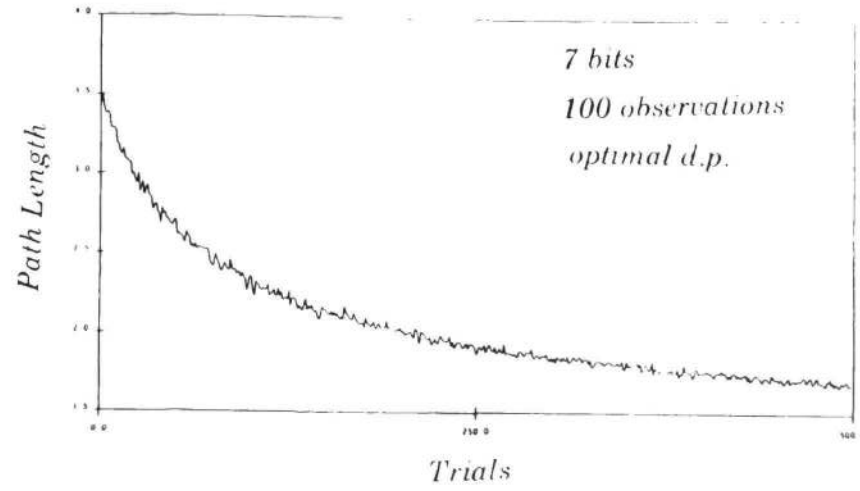
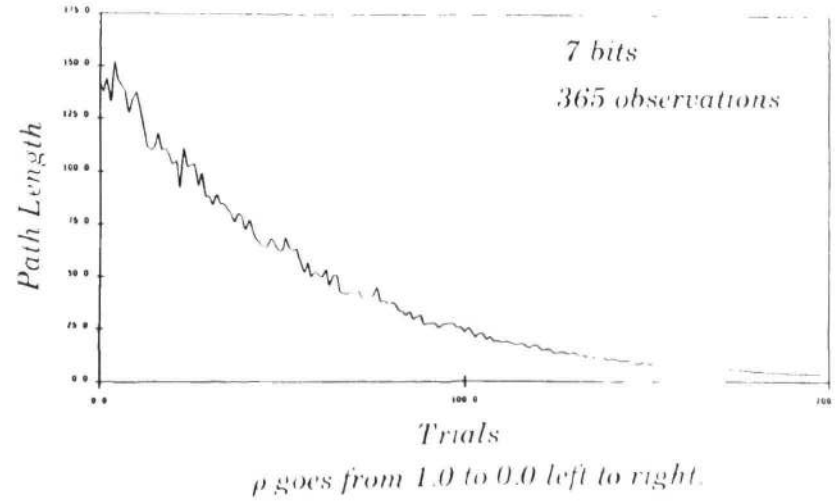


Figure 3: Bit Game power-law due to decision-procedure improvement.



ρ goes from 1.0 to 0.0 left to right.

ASSESSING THE STRUCTURE OF KNOWLEDGE IN A PROCEDURAL DOMAIN

Michel C. Desmarais, Luc Giroux, Serge Larochelle, Serge Leclerc

Université de Montréal and
Canadian Workplace Automation Research Center

In most domains of knowledge, the process by which someone learns to become more expert is relatively constrained. People learn the basic concepts before they learn the more complex ones. They learn the simple, even though inefficient, methods for doing things before they learn the efficient but more complex methods.

These constraints in the process of learning reflect the structure of precedence, or of increasing complexity among knowledge items. This structure can be regarded in terms of implications: the knowledge of some complex concept implies the knowledge of some other, more simple concept. Similarly, the usage of an inefficient method implies that another, more efficient method is *not* known.

Obviously, the structure of implications among knowledge items is of extreme importance from a pedagogical point of view. It dictates the order in which to teach those items. It is also of great importance for knowledge assessment, since it is by such a structure that a tutor can infer what is or isn't known, and test just the right knowledge items such that the result will yield the most information about the individual's state of knowledge.

Structures that represent precedence, or increasing complexity of knowledge items, are well known in education (see Tatsuoka, 1985) and have, in fact, already been used in human-computer interaction to automatically infer user knowledge of a system (Zissos & Witten, 1986; Chin, 1986). Formal properties of implication structures have also been investigated by Doignon & Falmagne (1985). So far, the main approach has been to construct these structures intuitively, based on someone's experience of how the knowledge items are interrelated or on some evaluation of knowledge item complexity. Another approach has been to use text-books, course content, or documentation to verify the ordering in which knowledge items are introduced and to consider this as a basis for the knowledge structure. For instance, Pavel (1985) constructed a knowledge structure from the automatic analysis of UNIX™ on-line documentation. Our approach, similar to that of Pavel (1985), is based however on empirical data (see Desmarais & Pavel, 1987, for a previous study with the current approach). We will show how to construct such a structure from data on a number of individuals' knowledge of a domain. This approach has the advantage of not being biased by subjective judgment about the precedence or complexity of knowledge items. Moreover, given data on a number of individuals' knowledge states, the whole process of knowledge structure construction can be automatized.

This paper thus presents an empirical assessment of the methodology for constructing knowledge structures from data on individuals' knowledge state. We will demonstrate how such structures can be build and how efficient they are for inferring a single individual knowledge state from partial knowledge of that state.

THE KNOWLEDGE STRUCTURE

The knowledge structure is composed of knowledge items. No constraints is imposed on the definition of knowledge items. They can represent the comprehension of conceptual information as well as the ability to perform some task. All that matters is that the knowledge items define the knowledge domain in some meaningful and complete way.

The knowledge items are related to one another by two types of binary relations: the *logical implications* (or simply "implications") which states that A implies B, and the *negative implications*, which states that A implies *not* B. Those relations form the knowledge structure in question, which we will call the *implication network*.

Logical Implications

Probably the most important types of relation for knowledge assessment are that of precedence and of increased complexity among knowledge items. Those will be represented by **logical implications** in the implication network, in so far as they permit the inference of items that are known or not. In other words, precedence or increased complexity from A to B corresponds to a logical implication from B to A, to the extent that we infer that A is known if B is, and that B is not known if A isn't. This is not to say that precedence, increased complexity, and logical implication are all the same thing, but simply that they have the same properties and we will treat them as interchangeable here.

Figure 1 illustrates, by means of implication relations, the interdependencies that may exist amongst the abilities to solve problems in mathematics. Notice first that the structure forms a minimal partial order (it contains no cycles and no transitive relations). Notice also that the relations can be of different nature. In some cases, as in $2 \Rightarrow 1$, the implication stems from the fact that the type of problem in 1 is found in 2 as a sub-problem (i.e. we find a division problem in the algebra problem 2). Thus we have a clear precedence from 1 to 2. However, the implications from 5 to 4 and to 3 bear no such precedence, since problem 5 belongs to graph theory and does not require any knowledge of calculus nor matrices. It turns out that the implication is due to the extreme complexity of the solution for 5 (this problem was first proposed in 1852 and solved 125 years later after great efforts by many mathematicians) which suggests that if you solved that problem you must be knowledgeable enough to solve 3 and 4.

Negative Implications

The second type of relations is a *negative implication*. As an example, consider the fact that a student laboriously solves a complex system of linear equations algebraically, when the usage of matrices would have been much more efficient. From this example, we can conclude that the student does not master the matrix method, and that there is a negative implication from the algebraic to the matrix method of solving a complex linear equations system. Note that in a negative implication, one must discriminate between usage and knowledge, for it is the *usage* of some knowledge item that implies that another knowledge item is not known. Naturally, this type of relation is found in domains where knowledge items are manifest in some performance, that is, in *procedural domains*.

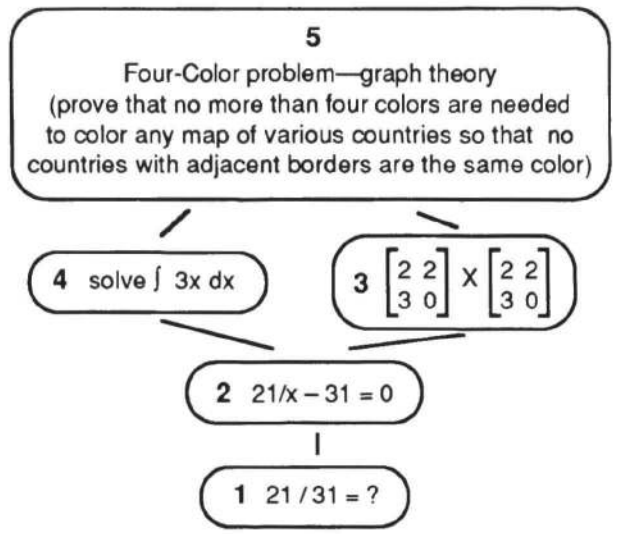


Figure 1. Implication network composed of logical implications among abilities to solve mathematical problems. Success for an item permits to infer that items implied are known, whereas failure permits to infer that items implying the failed item are not known.

Implication Network in a Procedural Domain

Although we have discussed the notion of implication structure in a general sense so far, we will now move the discussion in the context of *procedural knowledge domains*. Procedural domains can be characterized by *task structures* that define plans for completing specific tasks. Tasks also define **competences that can be represented by knowledge items**. Let us say a few words on the task domain in order to describe how the knowledge of that domain is represented and how the task structure relates to the implication network.

Text-editing

The domain in which we conducted our study is text-editing. This choice is determined by the fact that this study is part of a project for building an expert-system consultant for text editing (Desmarais, Larochelle, & Giroux, 1987). Thus, we are interested in the implication network both for pedagogical and for knowledge assessment purposes.

Text-editing is largely a procedural domain, in the sense that, in addition to concepts, the knowledge of this domain consists of actions, or procedures, for doing text-editing tasks. In fact, in the current study, **we will limit ourselves to the procedural**

dimension of text-editing, that is, to knowledge items that represent tasks, or *goals*, and to *primitive actions*, into which goals will ultimately be decomposed. (A primitive action is a task that cannot be further decomposed into sub-tasks and generally represents system functions, whereas a goal is a task that can be decomposed into sub-tasks, which can either be sub-goals if they are themselves further decomposed, or actions if they are not.) Thus, we will not have knowledge items that directly represent *concepts* like "buffers", or properties we can attach to characters (font, orientation, etc.) or to paragraphs (indentation, centering, etc), etc. Note, however, that such information could, indeed, be directly represented in the implication network and play an important role for knowledge inference.

Many tasks may be decomposed in a number of alternative ways which we will call *methods*. For instance, the task of replacing every occurrence of a word by another word can be achieved "by hand", replacing each occurrence one by one, or with some specialized system function designed especially for that task. The first method will thus be further decomposed into subgoals and actions for replacing text, whereas the second method will consists of a single primitive action (a system function).

Evidently, the second method will generally be much more efficient than the first one. In fact, given this information on method efficiency, we could presume that someone who uses the "find and replace each occurrence" method doesn't know the "search-replace" system function method. This leads us to a negative implication:

```
<find and replace each occurrence>
  ⇒¬ <search-replace function>
```

This relation is based on the postulate that if someone uses a sub-optimal method of doing some task, then the optimal method is not known. In a context where knowledge assessment is based solely on *known* knowledge items, as is the case of a coach who observes someone's performance, negative implications play a fundamental role for they are the only means of inferring what is *not* known from an implication network.

IMPLICATION NETWORK CONSTRUCTION

We described so far the nature of an implication network and its relation to the knowledge domain. In particular, we showed how some of the implications can be inferred from the task structure. We now turn to the problem of inferring *implications* and *negative implications* from **data on a number of individuals' knowledge state**.

Assessing Individuals' Knowledge State

The first step in the process of building an implication network is to gather information on a number of individuals' knowledge state. In order to do so, we elaborated a fairly exhaustive test to assess someone's knowledge of the editor WordPerfect™ (Leclerc, in preparation)[1]. The test contains 190 tasks. It is designed so that **the mastery of every major system function is tested individually by a task.**

There are three types of knowledge items that are associated with each task:

(1) **goal**: knowledge item that represents the task itself and which is considered mastered if the task is successfully completed, no matter what method is used;
(2) **methods**: knowledge item that corresponds to one or more primitive actions used *in the context of a goal.*
(3) **primitive action**: knowledge item that corresponds to a system function used; a primitive action is considered mastered if it is used successfully, no matter what the context is.

Hence, for each task, we find one knowledge item for the goal, one for each primitive action the task is decomposed into, and one for each method by which that task can be accomplished.

The distinction between a primitive action and a method enables us to discriminate between the usage of a system function in two different context. For instance, consider the two tasks of moving the cursor to the end of a word and to the end of the document. Although we will get a single knowledge item for the *primitive action* of moving the cursor one character to the right in both tasks, we will also get two different knowledge items for the *methods* which make use of that primitive action in each context. Indeed, moving the cursor to the right for going to the end of a word and for going to the end of the document legimately constitutes two different *competences*.

1 Leclerc, Serge (in preparation). Analyse de la structure de la connaissance des usagers d'un éditeur de texte, M.Sc. thesis in preparation, Département de psychologie, Université de Montréal.

Using the three types of knowledge items for the 190 tasks, we obtain the following distribution of knowledge items that compose the nodes of the implication network:

- 190 goals
- 195 primitive actions
- 286 methods

The total number of knowledge items is thus 671 (190 + 286 + 195).

The test was administered to 30 subjects. The performance varied from 56 to 146 successful tasks, which corresponds to 162 to 431 mastered knowledge items with an average of 307.

Compilation of Implications and Negative Implications

Once the data on individual knowledge states is gathered, the next step is to establish implications and negative implications among knowledge items. To determine if there is a relation between a pair of knowledge items, say A and B, we take the distribution of subjects along the following four situations:

(1) A and B are known
(2) A is known and B is unknown
(3) A is unknown and B is known
(4) A is unknown and B is known

then we state that there is an implication from A to B if we find people in situations 1, 3, and 4, but none in 2, as this situation would be impossible if, indeed, there were an implication. If there were a negative implication from A to B, then we would find people in all situations but 1. Establishing the implication network's relations then consists of analyzing the distribution of subjects over the four situations for each pair of knowledge items (the total number of pairs is 671 * 671 = 450,241).

Statistical parameters

If the world was black and white and there was an implication from A to B, we should never find a distribution like the following:

		B: *Known*	B: *Unknown*
A	*Known*	20	1
	Unknown	8	1

But because of noise, or simply because the implication does not reflect a clear precedence but something more like a strong surmise relation, we need some kind of statistical criterion to determine if there is or not a relation. We have used two statistical parameters to make this decision:

(1) the **minimal conditional probability** of B given A, paired with an **alpha error**. That is, given a measured conditional probability (20/21 = 0.95 in the distribution above) and a minimal conditional probability (we chose **P(B|A)>0.85**), we accept the implication from A to B if the lower bound of a [1 - alpha error] confidence interval around the measured conditional probability is greater than the minimal conditional probability (we chose an alpha error of **p<.20** which corresponds to an 80% confidence interval). In the distribution above, the lower bound of an 80% confidence interval around 0.95 is .86. Since it is greater than 0.85 we would accept the implication on this basis. However, even if we had a significant conditional probability, it is not necessarily different from the *initial* probability, meaning that the fact that A is known does not tell us anything more about B and that, consequently, there is no relation between A and B. For that reason, a second statistical parameter is needed.

(2) **the minimal probability of interaction, chi-square**: the chi-square value of the distribution determines if, indeed, there is a relation between two items. In the current study, we chose a chi-square corresponding **p<0.15**. For the distribution above, the chi-square is 0.41 and not significant at p<0.15. Thus we would reject the implication on the basis of this parameter.

Both parameters must be over the chosen criteria in order to set a relation, and both criteria apply to implication as well as negative implication relations (except that instead of 0.85 we will take 0.15 for the minimal conditional probability and look at the upper bound of the confidence interval).

Pruning and grouping

Having set the relations, it is often the case that we find transitive (A ⇒ B, B ⇒ C, A ⇒ C) and symmetric implications (A ⇒ B and B ⇒ A). Because transitive relations are redundant in the knowledge inference process, they are removed from the structure. As for knowledge items involved in a symmetric implications, they are grouped together to

Table 1
Distribution of the number of relations, nodes, and knowledge items in the implication network

	relations	nodes	K.I.
implications	2804	145	355
non transitive	*393*		
-incoming		77	203
-outgoing		137	289
-incoming & outgoing		69	137
negative implications	3247	346	555
-outgoing		145	354
-incoming		201	201
-incoming & outgoing		0	0

form a single node in the structure. All incoming and outgoing relations of each node in the group are redirected to that node. The resulting structure corresponds to a minimal digraph.

Composition of the Implication Network Derived

Of the 671 knowledge items to start with, only 555 are involved in the implication network and, consequently, 116 knowledge items (671 – 555) are not related to any other items (35 of those 116 are not related because they are known by every subject and thus are rejected by the *minimal probability of interaction* criterion: indeed, it is impossible to establish any interaction with another item in this case). Moreover, because of grouping, the 555 knowledge items only form 346 nodes in the structure. There are 97 groups involving between 2 and 6 knowledge items, and one involving 47 knowledge items (grouping often occurs because a goal is always achieved by a single method, in which case the goal and the method will mutually imply each other); 248 nodes are composed of a single knowledge item. The distribution of implications and negative implications is given in table 1.

The structure is composed of a total of 2804 implications. 137 nodes have outgoing implications, 77 have incoming and 69 have both incoming and outgoing implications. Taking into account nodes that comprise multiple knowledge items, those figures become 289, 203, and 137 knowledge items respectively. Of the 2804 implications, only 1220 are non redundant (i.e. are not different paths between the same nodes) and only 393 are non transitive (i.e. form the minimal digraph). The distribution of the number of intermediate nodes in the transitive implications is compiled in table 2.

Table 2
Distribution of intermediate nodes in transitive implications

Intermediate nodes	Frequency
1	718
2	790
3	579
4	239
5	69
6	16

As for the negative implications, they are much more numerous with 3247 relations, but there is no transitivity with this kind of relation. All the 346 nodes in the implication network are involved in negative implications. 145 nodes have outgoing negative implications (355 knowledge items) and 201 have incoming negative implications (all of which are single knowledge item nodes).

VALIDATION OF THE IMPLICATION NETWORK

How valid are the implications derived? To answer this question, we conducted a simulation of knowledge inference for each of the 30 subjects with the following procedure: we sampled randomly a portion of the **successfully** completed tasks of a subject and fed this information to the knowledge inference module, which inferred the known and unknown knowledge items according to the implication and negative implication relations, and from the grouping of knowledge items. (Because of the context of our research, we wish to simulate the situation of a coach which is restricted to infer a pupil's knowledge from the *observation* of competence—a coach does not have direct information on what isn't known). We then compared the inferred known and unknown knowledge items with the actual knowledge state of the subject as measured directly from the test.

The results of the simulation are summarized in figure 2. A compilation of the correct ("hits") and incorrect ("false alarms") inferences is plotted for both known and unknown knowledge items, as a function of the proportion of the subjects' successful tasks that was sampled and fed to the knowledge inference module. The graphs show the values averaged for the 30 subjects. The white area indicates the size of the sample; the grey area indicates the information added by the inference process (hits); and, finally, the black area indicates the incorrect inferences (false alarms).

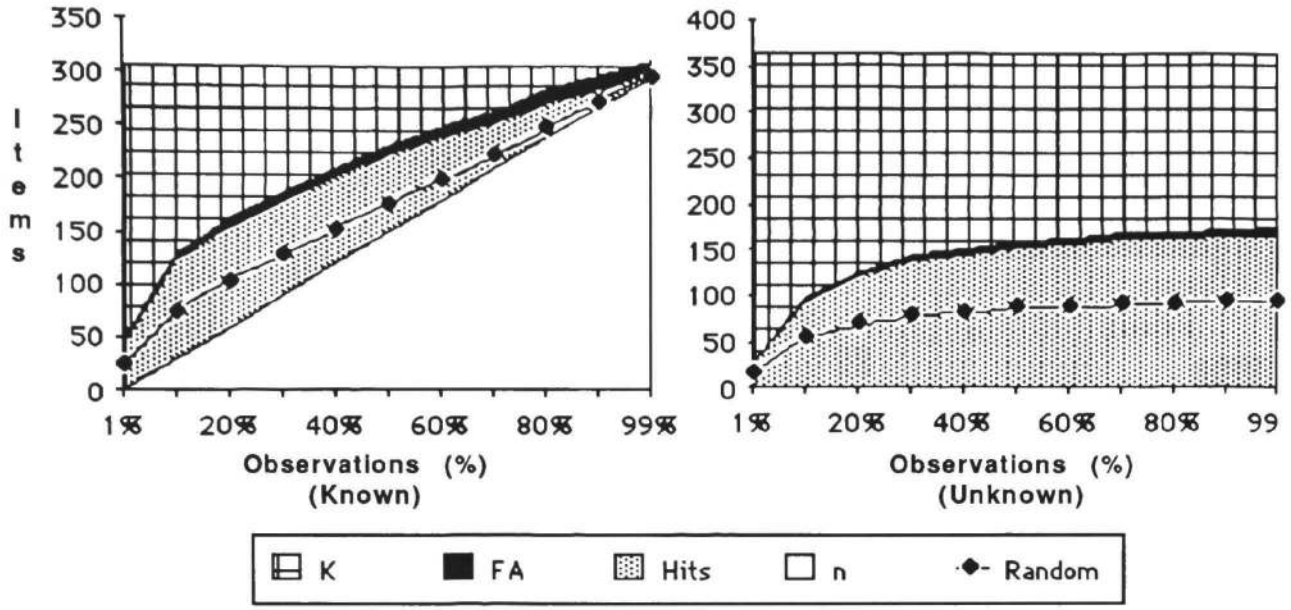


Figure 2 Analysis of the inference of known and unknown knowledge items as a function of the proportion of known items given to the knowledge inference module. The area labeled 'K' represents the actual known and unknown knowledge items. The dark area represents incorrect inferences (False Alarms) whereas the grey area represents the correct inferences, or 'hits'. 'n' is the number of knowledge items given. Hence the grey area represents the correctly inferred information for knowledge assessment. We included a "random inference" curve as a comparison.

The proportion of false alarms is relatively low in both graphs (below 10%), but increases for the known items to the point where all inferences are false alarms when the inference module is fed with *all* of the successful tasks, as can be expected. In fact, the greatest proportion of "added information" is between 10% and 40%, where the unknown inferred knowledge items are close to their maximum and where the proportion of false alarms over the hits is still relatively low for the known knowledge items.

Initial vs. added information

It must be noted that, for the purpose of knowledge assessment, the knowledge items inferred on the basis of the implication network constitute "added information" to what we ought to call the "initial information". That is, if we were to make a knowledge assessment from a sample of a subject's knowledge state, we would start with the initial information, namely, the initial probability of knowledge, and add to it the knowledge inferred from the implication network. For instance, in our case, we would start with at least 35 items known since their initial probability is 1 (they were known by every subjects).

The more severe the *minimal probability of interaction* is, the more the added information will differ from the initial information. In other words, the minimal probability of interaction assures us that the inferences made constitute information we didn't start with.

CONCLUSION

We have demonstrated that we can establish implications and negative implications among knowledge items, as well as grouping, by means of an empirical method and that we can characterize the structure constructed by some statistical parameters. We have also demonstrated that this structure can be used to infer a portion of known and unknown knowledge items from the observation of a portion of the known items. Moreover, we have all reasons to believe that with a sufficient number of individual knowledge states we can capture the structure of implication among knowledge items and that this structure constitutes a fundamental dimension of knowledge.

On the other hand, a number of questions remains unanswered. Although the simulation showed that the

structure has a relevant power for knowledge inference, it is not clear how much more efficient it is compared to other, more simple schemes, like a linear structure where we simply order knowledge items as a function of the number of people who know them. If, in fact, the structure of the knowledge domain was linear, the performance of the current model (in terms of knowledge inference) would turn out to be similar to that of model based on a linear structure, as, indeed, the structure derived would itself be linear.

It would also be interesting to know precisely how many implications we missed with the small number of subjects we had and how many would be required to miss only a few (hopefully there might be a computable answer to this question from the statistical parameters of the network).

We are currently in the process of comparing the present knowledge inference scheme to simpler ones. A qualitative analysis of the structure obtained is also under way (Leclerc, in preparation). Finally, we have plans to repeat this experiment with other knowledge domains.

REFERENCES

CHIN, David N. (1986). User Modeling in UC, the UNIX Consultant, *Human Factors In Computing Systems, CHI'86 Conference Proceedings*, 24-28, Association for Computing Machinery, New York: NY.

DESMARAIS, Michel C., LAROCHELLE, Serge, & GIROUX, L. (1987). The Diagnosis of User Strategies, *Human-Computer Interaction—INTERACT'87*, H.-J. Bullinger and B. Shackel (Eds.), Elsevier Science Publishers B. V. (North Holland), 185-189.

DESMARAIS, Michel C. & PAVEL, Michael (1987). User knowledge assessment: An experiment with UNIX, *Human-Computer Interaction—INTERACT'87*, H.-J. Bullinger and B. Shackel (Eds.), Elsevier Science Publishers B. V. (North Holland), 151-156.

DOIGNON, Jean-Paul & FALMAGNE, Jean-Claude (1985). Spaces for the assessment of knowledge, *International Journal of Man-Machine Studies*, **23**, 175-196.

PAVEL, Michael (1985). UNIX knowledge assessment, Psychology Department, Stanford University.

ZISSOS, Adrian Y. & WITTEN, Ian H. (1985) User modelling for a computer coach: a case study, *International Journal of Man-Machine Studies*, **23**, 729-250.

TATSUOKA, Maurice M. (1986). Graph theory and its applications in educational research: A review and integration, *Review of Educational Research*, **56**, 291-329.

IMPROVEMENT IN MEDICAL EXPERTISE INDEPENDENT OF STABLE KNOWLEDGE

G.R. NORMAN, L.R. BROOKS, S.W. ALLEN, D. ROSENTHAL
McMASTER UNIVERSITY, HAMILTON, ONTARIO

An intuitively plausible position about the acquisition of expertise is what we will call the Independent Cues interpretation: learners gain expertise mainly by acquiring knowledge about the specific features (signs or symptoms) which characterize a disease or condition and those features which are best able to differentiate among diseases. (The term independent cues follows Smith & Medin, 1981 in their classification of concept theories). This model of learning is the implicit, if not explicit, goal of most instruction in clinical diagnosis. This assumption also underlies models, such as Bayesian or regression decision models that capture increasing expertise with changes in weights of features.

One consequence of an independent cues model is that performance should improve more rapidly on typical than atypical cases. Since typical cases possess more of the features which are characteristic of a category, these should be mastered with relative ease. Conversely, atypical cases have few features in common with a category, hence would require a high level of expertise to differentiate from other conditions. In addition, cases which are empirically relatively easy for neophytes should be mastered with perfect accuracy by experts.

In this paper we will concentrate on the relative performance on typical and atypical, easy and difficult cases of clinicians at three different levels of expertise in dermatology. The task we studied was the diagnosis of common skin disorders on the evidence provided by color slides, some of which were judged typical of the represented disorder and others were atypical.

EXPERIMENT 1

Method

Six subjects were chosen at each of three levels of expertise in dermatology: first year residents in family medicine, general practitioners, and practicing dermatologists.

The stimulus materials were 100 slides chosen from the slide collection of an academic dermatologist. Five slides were chosen from each of 20 common skin conditions, with two judged by the dermatologist to be typical presentations and three atypical presentations. A brief history, consisting of 1 to 4 lines of typed text and intended to be typical of the disorder was created by the dermatologist for each slide. Since this variable is irrelevant to the focus of the current paper, all error analyses were collapsed across history. The slides were presented in a randomized series, using four different starting positions to

balance for order. For half the items, balanced across subjects, the subject first read the history and then viewed the slide. Subjects were asked to diagnose each case as rapidly as possible or to indicate "don't know."

Results

Mean Errors. The average error rate for the groups were Residents = 44%, General Practitioners = 33%, Dermatologists = 14.5%, resulting in a highly significant effect of expertise on mean errors (chi-squared = 1337, p=.0001). The use of the "Don't Know" option was minimal, ranging from 7% for residents to 0.3% for dermatologists.

Response time

Response times were separately calculated for correct, incorrect, and 'don't know ' slides. The mean response time for correct identifications declined slightly with expertise, from 9.0 sec. for residents to 7.7 sec. for dermatologists. By contrast, errors for all groups were significantly slower than corrects, and increased significantly with expertise (residents 12.2 sec.; general practitioners 15.3 sec.; and dermatologists 17.5 sec.; F=10.9, p<.001). The positive association with expertise was even more pronounced for the 'don't know' slides; residents took an average of 19.3 sec., general practitioners 24.4 sec., and dermatologists 26.3 seconds (F=3.31, p<.05). These results suggest that errors do not apparently result from carelessness or speed and lack of thoroughness; if anything, the converse appears to be true.

Typicality.

To assess the effect of typicality, the number of errors was first corrected for frequency (there were 2 typical and 3 atypical cases per disorder) and then formed into the ratio of errors on typical items divided by total errors. An equal tendency to make errors on typical and atypical items would result in a .5 value for this ratio, and learning to deal effectively with typical items before atypical items would result in declining values with expertise. In fact, the proportion of errors made on items designated as typical by the dermatologist were approximately constant over the three levels of expertise, despite the threefold decrease in overall errors (R=.40, GP=.42, D=.40).

Average Item Difficulty.

An independent cues model implies that i) the difficulty of a case is related to the degree to which the cues present in the item support a single diagnosis, and ii) expertise is related to the knowledge of the appropriate combinations and weightings of

these cues. Thus if we consider items which are empirically easy or difficult for residents, most improvement with expertise should arise on easier items. Conversely, some ambiguous slides are likely to contain insufficient information for accurate diagnosis, and these should show little improvement with expertise.

An alternative position is that errors of clinicians are a result of carelessness or inattention. If this were the case, there should be no association between the difficulty of an item, based on the performance of residents, and errors committed by dermatologists, since errors result from a random process unrelated to any measure of item difficulty.

To explore these models, we characterized the difficulty of each slide on the basis of the errors committed by residents, thus an easy slide had 0/6 errors by residents, a difficult slide had 6/6 errors by residents, and there were 5 intermediate levels of difficulty. We then examined the proportion of errors at each level of expertise committed on slides at each level of difficulty.

Because the difficulty of slides is based on the performance of residents, the plot of resident errors at each level of difficulty is a straight line through the origin. From the independent cues model, we would anticipate that as expertise is acquired, proportionately more errors will be committed on difficult slides, so that the distributions move to the right with expertise. Conversely, if errors were a result of random processes such as inattention, the likelihood of an error by G.P. or dermatologist should be unrelated to the resident item difficulty, and the distribution should be flat.

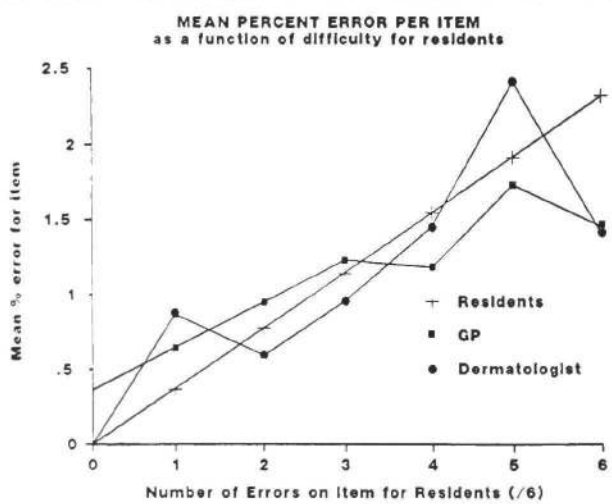


FIGURE 1
ERROR RATE OF RESIDENTS, G.P.'S, AND DERMATOLOGISTS RELATED TO ITEM DIFFICULTY

The results are shown in Figure 1. It is apparent that the distributions for general practitioners and dermatologists are similar to those of residents, i.e. an approximately straight line with positive slope. More important, although we would predict from an independent cues model that, with increasing expertise the curve would shift to the right, the data provide no evidence of this shift.

Thus, although the absolute error rate declined by about a factor of three from resident to dermatologist, we found no evidence that expertise resulted in improvement on relatively easy or typical items.

EXPERIMENT 2

The analysis of typicality from the first study is subject to possible idiosyncracies in the ratings of typicality since the categorization was done by only one dermatologist and consequently might be subject to some unreliability. Also, perceived typicality itself might vary across expertise groups as well as having an uncertain relation to the basis of diagnostic performance. Therefore, it is critical to determine if the results of this study held up when the typicality ratings of another group were substituted.

The second study addressed some of these issues. In the context of another study examining the effect of prior exposure on ratings of plausibility (the hindsight effect), we obtained ratings of typicality on a total of 69 of the slides used in the present study from a varying number of general practitioners ranging from 3 to 6. Following the initial presentation, a second session was arranged with each subject a minimum of 4 weeks later. At this followup session, subjects were shown a total of 32 slides, of which 16 had been used in the first session and 16 were new (counterbalanced across subjects), and were asked to diagnose the conditions. The second session permitted an examination of the effect of a single prior exposure on diagnostic performance.

To conduct this analysis, the proportion of the general practitioners rating each slide as typical was determined. Five levels of categorization were created: typical by total agreement (22 slides), typical by majority agreement (25 slides), equally divided (3 slides), atypical by majority agreement (13 slides), and atypical by total agreement (6 slides). Although it is evident that G.P.'s were more inclined to rate slides as typical than the original classification of the dermatologist would indicate. nevertheless there was reasonable agreement between the two sources. Of the 23 slides originally rated as typical by the dermatologist, 21 (91%) were rated typical by a majority of G.P.'s. However, only 16 of the 44 slides (36%)initially classed as atypical were so rated by a majority of G.P.'s.

An analysis was then conducted as before, calculating the proportion of errors made on typical and atypical slides, considering both slides on which there was total agreement and slides where a majority, or all, G.P.'s agreed. The results are shown in Table 2 below:

Table 2
Proportional Error Rate by Expertise

	Resident	General Practitioner	Dermatologist
Total Agree	.339	.345	.343
Majority Agree	.338	.395	.339

It is evident from this table that the constant proportionality of errors on typical slides as a function of expertise is also evident with the G.P. ratings, thus it does not appear to be a result of the idiosyncratic categorization of the dermatologist.

As one final converging evidence on the topic, we examined the performance of G.P.'s in diagnosing slides which they themselves had previously rated as typical or atypical. Average error rate on self-rated typical slides was 29%, and on atypical slides was 54%, for a ratio of .34, consistent with the ratios shown in the previous table.

Thus, it is apparent that although performance of all groups was better on slides judged as typical by a variety of approaches, there was no association between improvement in performance related to expertise and typicality. Put another way, slides judged as typical presented just as much diagnostic difficulty (proportionately) to dermatologists as to residents.

Prior Examples and Diagnosis

It is apparent that the independent cues model and a model which views errors as a random event fare poorly in accounting for the improvement in expertise observed in the present study. An alternative model of concept formation (Brooks, 1987) postulates a central role of prior instances in recognition.

The second study provides a partial test of this model. In addition to examining relative error rates in the second session on those slides rated by each G.P. as typical or atypical, we also examined the error rate on slides which had been seen previously in the context of the typicality-rating task and a balanced set of new slides. The results are as shown below:

Table 3

Error Rates on Old and New Slides

	Previously Seen	New
Typical	31%	46%
Atypical	48%	60%

Thus, a single prior exposure to the slide, a minimum of four weeks previously, resulted in a 20-30% reduction of error rates. These data suggest that the diagnostic task may be strongly influenced by recall of prior instances of a category.

DISCUSSION

In these data the traditional indices of category structure - typicality and average item difficulty, are roughly constant over a large range of accuracy. We conclude that the improvement over the range of expertise observed in this study is not a matter of learning items in order of difficulty or learning more appropriate weights for the essential symptoms and signs. In other words, these data are incompatible with an independent cues interpretation of acquisition of expertise. In fact, the observed constant proportionality rules out any model that determines typicality, average item difficulty, and improvable error by the same information.

This finding also provides difficulty for stable instance-based models of categorization (e.g. Hintzman 1986, Brooks, 1978), which would hypothesize constant availability of prior instances of a category, since in this view the expert has available a relatively stable array of prior instances of a category, and diagnosis is conducted by a comparison of similarity to available instances. The difficulty of this model is that there should be many more typical instances available to the expert for similarity judgement, thus expert performance should be proportionately higher on typical and easy slides.

By contrast, although the data suggest a role of prior instances in expertise, what makes a previous instance available is not just a fixed set of features, but contextual information relating to how a prior instance was processed. Previous items in encountered in the same context are more available then items processed in a different context (Godden and Baddeley, 1975). Similarly,items treated in a similar manner are more available than those processed differently (Cermak and Craik, 1979).

How could this kind of variability in the availability of prior instances provide an explanation of the observed data? If

access to prior instances is context-dependent this could affect overall performance without necessarily changing the relative difficulty of typical and atypical, easy and hard items. For example, the previous occurrence of particular diagnoses in a series may result in the increased availability of that category, hence a diagnostic bias. A second possibility is that certain contextual factors, such as the location of the lesion or the physical appearance of the patient might result in bias in favour of a particular diagnosis and result in errors which are unrelated to objective categorizations such as difficulty or typicality. The contribution of such error factors could be expected to decline with increasing expertise.

Obviously, we are not claiming that there is no such thing as "independent cues" knowledge of the features of particular diseases, or that such knowledge is irrelevant to expertise. Rather, these data constitute a case in which there is massive improvement that is independent of such knowledge. It is entirely possible that the acquisition of the basic definitional, "independent cues" type of knowledge is restricted to the initial stages of learning, and the substantial change while obtaining practical experience is due to the retrieval factors just discussed. We further conjecture that these findings might be a common feature of any field in which the challenge is to recognize any of a large number of disorders occurring in a mixed series. Perhaps in a mixed series the effect of difficulty of an item is small by comparison to the effect of processing variations induced by the series itself. If so, then the emphasis in training in such areas should be on providing practice with mixed series and with emphasizing to the learners the importance of mixed practice for developing expertise.

REFERENCES

Brooks L.R. Non analytic concept formation and memory for instances. In E.Rosch and B.LLoyd (ed), Cognition and Categorization. Hillsdale N.J., Lawrence S. Erlbaum, 1978.

Brooks L.R. Decentralized control of categorization: The role of prior processing episodes. In U. Neisser (ed), Concepts and Conceptual Development: Ecological and Intellectual Factors in Categorization, New York, Cambridge University Press, 1978.

Cermak L.S. and Craik F.I.M. Levels of Processing in Human Memory. Hillside N.J., Lawrence S. Erlbaum, 1979.

Godden D.R. and Baddeley A.D. Context dependent memory in two natural environments on land and underwater. Brit. J. Psychol. 66, 325-332, 1975.

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

Sequential Connectionist Networks for Answering Simple Questions about a Microworld

Robert B. Allen
Bell Communications Research

Sequential back-propagation networks were trained to answer simple questions about objects in a microworld. The networks transferred the ability to answer this type of question to patterns on which they had not been trained. Moreover, the networks were shown to have developed expectations about the objects even when they were not present in the microworld. A variety of architectures were tested using this paradigm and the addition of channel-specific hidden layers was found to improve performance. Overall, these results are directed to the approach of building language users with connectionist networks, rather than language processors.

Introduction

Because neural algorithms such as back-propagation [9] are such effective techniques for machine learning, it is now possible to seriously consider developing systems which learn to *use* language. Moreover, neural networks have many other characteristics which make them especially suitable for such an effort. They are sensitive to context, they can adapt to exceptions, and input from diverse sources can be easily combined. The concept of developing a language user is an alternative to the usual approach in artificial intelligence of attempting to process the components of language and then to synthesize an "understanding" from those components. Rather, the approach suggested here is to train networks under a broad enough range of conditions to be able to understand and respond with language-like stimuli. Potentially, this approach may provide a robust basis for linguistic processes ranging from translation [1] to speech recognition. Of course, the extent to which these networks can be said to actually 'have' language may well be as difficult and controversial as the evaluation of linguistic capabilities of apes (see [7]).

In the research described here, sequential networks were trained to accept language-like stimuli which refer to objects in the microworld. This paradigm is based on the assumptions that language is most readily acquired through interaction with the world [1] and that language learning is essentially a supervised learning process. Inputs of two types were employed, a coded microworld and sequential verbal codes, which formed questions about the microworld. One type of sequential network which might be applied to this task is shown on the left of Fig. 1. This network, which may be termed an output-feedback network, was developed by Jordan [6] to model articulation. The output units from one cycle feed 'state' units which are used as extended inputs for later cycles. As shown on the right side of Fig. 1 a variation of that procedure, suggested by Elman [4], draws feedback from the hidden layer rather than the output layer. Fig. 2 presents a sequential network which may be termed a generalized hierarchical-sequential network. In this network inputs are divided into separate channels and have extra hidden layers for each of the sets of input units, as well as for the state units. This architecture has the advantage of being modular, hence it might be readily adapted to multi-processor computers and perhaps specialized perceptual hardware.

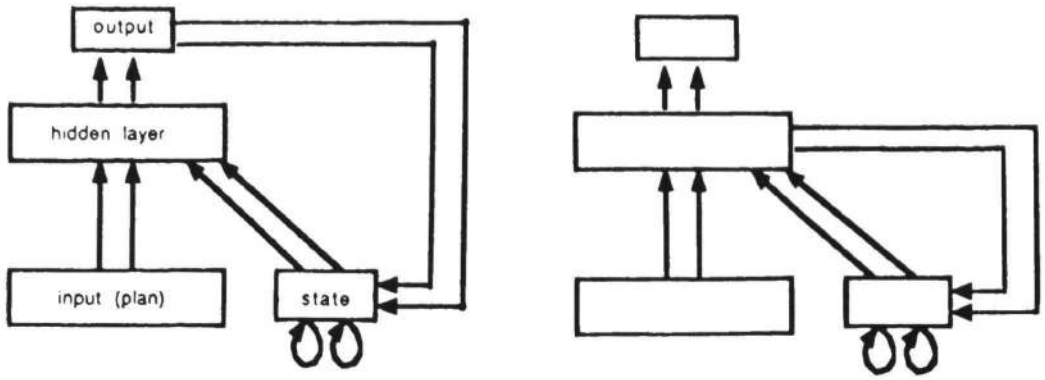


Fig. 1. Simple sequential networks with feedback from output and hidden units.

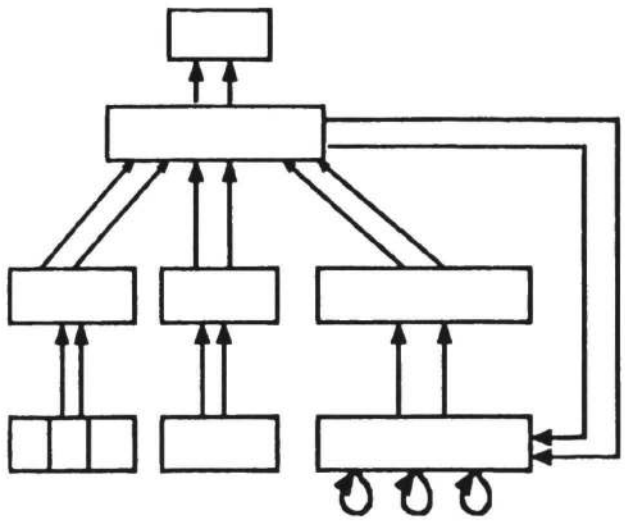


Fig. 2. Generalized hierarchical-sequential network.

Procedure

Coding

The verbal inputs were composed from questions from a vocabulary of 27 terms which were coded with randomly assigned 6-bit -1/1 codes. There were 16 output terms and their coding was randomly selected from 5-bit 0/1 codes. One set of inputs presented static codes which were characterized as objects in a 'perceptual' field. The perceptual field was composed of 3 slots, each slot consisting of 5 bits, in which any one of 8 objects could appear. 3 of these bits encoded the objects themselves and two additional bits coded features. The objects themselves were coded with randomly selected -1/1 codes, while empty slots were filled with nulls. In addition to the 3 bits which uniquely specified the objects, two additional bits were correlated with each object in the proportions shown in Table 1. While the probabilistic features will be more difficult to learn than perfectly correlated features, they are necessary to guarantee that the network attends to the microworld. On one hand, these bits might be considered to be an explicit feature

such as color. For instance, it would be possible to say that **object$_1$** had **color$_1$**. Alternatively, these bits could be thought of as context bits. For instance, some objects are highly correlated with places, while other objects are less likely to be correlated. The additional bits were associated with the objects in the following proportions:

object	feature	
	A	B
1	.9	.9
2	.8	.9
3	.7	.9
4	.6	.9
5	.4	.1
6	.3	.1
7	.2	.1
8	.1	.1

Table 1. Probability of object tokens having features A and B.

Question Construction and Sequential Presentation

The sequential presentation of verbal information is illustrated in Table 2. In the example two objects are in the perceptual field, **object$_2$** in **slot$_1$** and **object$_1$** in **slot$_2$**. Across the four intervals, the coded forms of the words **is**, **this**, **object$_1$**, and **fA$_1$** are presented. Because the input is sequential, correct output responses were generally not known until the question was complete. Before the correct output is known error values of zero were back-propagated. (A similar procedure in which there was no back-propagation on those trials produced similar, but slightly worse results.) These cycles are indicated by *** in the table and termed *don't care cycles* after the *don't care units* of Jordan.

interval	perceptual input			verbal input	verbal output
1	$o_2/fA_2/fB_1$	$o_1/fA_1/fB_1$	null	is	***
2	$o_2/fA_2/fB_1$	$o_1/fA_1/fB_1$	null	this	***
3	$o_2/fA_2/fB_1$	$o_1/fA_1/fB_1$	null	$object_1$	***
4	$o_2/fA_2/fB_1$	$o_1/fA_1/fB_1$	null	fA_1	yes

Table 2. Sequences of codes for typical question answering procedure.

Input/output patterns were prepared from the templates shown in Table 3. With the exception of a few very simple commands (e.g., repeat, describe), the input patterns were questions. A question such as **What fA is the objX$_1$** might be read, with nouns inserted, as **What color is the car?** While the answer, **fAX$_1$**, might be read **blue**. Clearly, the templates are somewhat ad hoc and stylized. The questions were associated with appropriate perceptual inputs; thus, the 8 objects could appear in 3 slots of the perceptual field. In cases with two objects in the perceptual field and only one of them was necessary to answer the question, the second object was randomly selected and its features were constrained so as not to conflict with the question. When questions could be answered yes/no, equal numbers of yes and no questions were prepared. 3242 unique inputs were generated, and from this set 25 were randomly chosen for

transfer.

verbal input	verbal output
object absent	
repeat $objX_1$	$objX_1$
repeat fAX_1	fAX_1
repeat fBX_1	fBX_1
one object present	
repeat $objX_1$	$objX_1$
repeat fAX_1	fAX_1
repeat fBX_1	fBX_1
what do you see	$objX_1$
describe what you see	$objX_1$
is this the $objX_1$	yes/no
do you see the $objX_1$	yes/no
is this $objX_1$ fAX_1	yes/no
is this $objX_1$ fBX_1	yes/no
what fA is the $objX_1$	fAX_1
what fB is the $objX_1$	fBX_1
what is the fA of the $objX_1$	fAX_1
what is the fB of the $objX_1$	fBX_1
is the $objX_1$ in the slotS1	yes/no
two objects present	
is the $objX_1$ fAX_1	yes/no
is the $objX_1$ fBX_1	yes/no
which is fAX_1	$objX_1$
which is fBX_1	$objX_1$
what fA is the $objX_1$	fAX_1
what fB is the $objX_1$	fBX_1
what is the fA of the $objX_1$	fAX_1
what is the fB of the $objX_1$	fBX_1
is the $objX_1$ in the slotS1	yes/no
is the $objX_1$ over/under the $objX_2$	yes/no
which is over/under the $objX_1$	$objX_2$

Table 3. Input/output templates.

Network Parameters

The weights from the hidden units to the state units were fixed at 1.0 and the self-weights on the state units were 0.5. All of the other weights were adaptive with η=0.01 and α=0.9. The networks described below were trained for 200K pattern presentations. Except as noted below, the simple networks had 15 perceptual units, 6 verbal input units, 50 state units, 50 hidden units, and 5 output units. In addition, the hierarchical networks had 15, 6, and 50 units in the perceptual, verbal, and state-hidden layers respectively. At the beginning of each question the state units were reset to zero; tests demonstrated that learning occurred without reset, although it was faster and better with resets.

Results

The transfer set consisted of 25 questions and because each question required a one word (5 bit) response, a total of 125 bits had to be generated. The sequential network with feedback taken from the output (left side of Fig. 1) made errors on 29 bits (12 words). The hidden layer feedback network (right side of Fig. 1) performed somewhat better, with 23 bit errors and 9 word errors. Indeed, more complex networks (e.g., the PVS, see Table 4) made as few as 6 bit errors and 2 word errors (see below). It is perhaps remarkable that these networks can learn this task because they rarely get explicit training for storing words in the early part of the sentence. For instance, in questions such as the one shown in Table 2, the network has to remember that the question concerns **object$_1$**.

Architecture Manipulation

The hierarchical-sequential network (Fig. 2) can be thought of as a family of networks which may be tested separately. The errors for the 8 possible networks (formed by all possible combinations of the presence/absence of the perceptual, verbal, and state hidden layers) is shown in Table 4. As a short notation, these networks may be referred to with three-letter codes for instance, a PNS network would have a perceptual hidden layer, no verbal hidden layer, and a state-hidden layer. The NNN network is the network shown at the right side of Fig. 1. All of the other networks perform better than the NNN network, and the best is the PVS network.

	N		S	
	N	P	N	P
N	23(9)	7(4)	17(8)	12(6)
V	18(6)	14(5)	14(7)	6(2)

Table 4. Bit errors (word errors) for different networks.

Additional tests showed that these results replicated, essentially following the pattern in Table 4. However, with other data sets the PVS network is not consistently found to be best. Moreover, considerable caution must be exercised in comparing the different architectures in Table 4 because they include different numbers of neurons and weights. As a control for this, several other networks were tested. First, an output feedback network (left side of Fig. 1) with 100 hidden units was tested; this performed relatively poorly, 26(13). As a second test, a NNN network with 65 units in both the hidden and state layers was tested, this had 13 bit errors and 9 word errors.

Several other architectures, related to hierarchical-sequential network described above, may also be considered. For instance, a network was investigated in which the state units and a state hidden layer were attached to the verbal hidden layer. With 30 hidden units, 20 verbal hidden units, 20 state units, and 20 state-hidden units, this network performed the task with 11 bit errors and 6 word errors.

Semantic Memory and Categorization

Previous connectionist research on semantic memory [5, also Rumelhart unpublished] may be extended by considering semantic memory in this paradigm with explicit verbal training. Thus, the PVS network trained above was presented with the question **What fA/B is the object$_1$?** when the object was absent from the perceptual field. The decoded responses are shown in Table 5, where 1 or 2 indicates one of the features belonging to that feature type, and X indicates an apparently meaningless answer. For 7 of the

objects the network learned the correct values of **fB**, which is more consistently associated with the objects than **fA** (see Table 1). However for **fA**, the network consistently assumed that all objects had one feature value with the exception of $\mathbf{object_1}$, which caused an error.

object	feature	
	A	B
1	X	X
2	2	1
3	2	1
4	2	1
5	2	2
6	2	2
7	2	2
8	2	2

Table 5. Responses to feature questions with the object absent.

Additional tests have demonstrated that networks can learn about features although the features are never presented in the microworld. Moreover, the networks have been found to readily learn category names for grouping objects. On the other hand, a network in which the microworld was entirely absent showed only a small improvement above chance performance.

An Extended Generalization Test

When working with complex stimuli such as the questions and microworlds used here it is possible to consider generalization at many levels. While the transfer test described above consisted of randomly selecting test cases from a corpus in which the same question was often asked about several different configurations of the perceptual field, the test described here completely dropped training on one question and then tested that question during transfer. Specifically all 41 questions were removed from the corpus which asked whether $\mathbf{object_1}$ had $\mathbf{fA_1}$. Although other questions asked whether $\mathbf{object_1}$ had $\mathbf{fB_2}$ and whether other objects had $\mathbf{fB_1}$. A PVS network trained on the patterns which were not deleted, transferred quite well with 6 bit errors and 4 word errors.

Discussion

Networks were shown to learn and transfer the ability to answer questions in which coded 'verbal' questions and objects are presented sequentially. Moreover, evidence was presented for the utility of hierarchical-sequential networks. Naturally, this research may be extended in many ways. For instance in the sequential verbal input paradigm, linguistic constructs such as plurals or negation could readily incorporated. Indeed [8] reports the comprehension of pronouns which refer to objects in the microworld. In addition, the contribution of the hidden layers might be investigated through either analysis of the activations or parametric manipulation of the numbers of units. Presumably the additional hidden layers in the hierarchical network transform the input encoding to an encoding which is more easily intergrated with the other sources of information. In the case of the perceptual inputs this suggests that language can affect perception, in other words a type of *linguistic relativity*.

The research described here has focused on the task of answering questions as a means of generating feedback for language training. While this may seem restrictive at first, it is possible to imagine many variations in which the training would be less explicit. For instance, the questions might not be directed

to the network. If the question were posed to a different agent which made the response, the agent which is acquiring language might learn pairings of input and output from observation and perhaps imitation of the other agent. Furthermore, aside from answers to explicit questions there are many types of feedback for language use such as correctly completing a verbal command or instruction.

Finally, the strategy of developing language users which interact with a microworld may be extended beyond sequential verbal inputs. Additional work is underway in which verbal and microworld information is combined to produce language-like behavior, for instance networks which generate sequential outputs and manipulate the microworld [2] and multiple networks which communicate to complete tasks [3].

REFERENCES

1. Allen, R.B. Several studies on back-propagation and natural language. *Proceedings of the International Conference on Neural Networks* (San Diego, 1987), II/335-II/341.
2. Allen, R.B. Generation of verbal descriptions and action sequences with connectionist networks. submitted.
3. Allen, R.B. and Riecken, M.E. Interacting and communicating connectionist agents. *Proceedings of the International Neural Network Society* (Boston, 1988).
4. Elman, J.L. Finding structure in time. UCSD-CRL TR#8801.
5. Hinton, G. Learning distributed representations of concepts. *Proceedings of the Cognitive Science Society* (Amherst, MA, 1986), 1-12.
6. Jordan, M.I. Attractor dynamics and parallelism in a connectionist sequential machine. *Proceedings of the Cognitive Science Society* (Amherst, MA, 1986), 531-546.
7. Premack, D. *Gavagai!* (MIT Press, Cambridge, MA), 1986.
8. Riecken, M.E. and Allen, R.B. Anaphora and reference in connectionist language users. submitted.
9. Rumelhart, D.E., Hinton, G.E., and Williams, R.J., Learning internal representations by error propagation. In: D.E. Rumelhart and J.L. McClelland (Eds.), *Parallel Distributed Processing*. (vol. 2). 1986, 318-362.

The usefulness of the script concept for characterizing dream reports

Daniel Deslauriers & George W. Baylor

Département de psychologie, Université de Montréal

A dream is usually defined as a mental experience that is (1) multimodal sensory imagination, largely visual; (2) dramatic in form, and (3) experienced as reality rather than imagination, i.e., hallucinatory. The dream itself is a subjective experience so that what is recalled and usually referred to as a dream is a dream report. The highest recall of complete, full-length dreams follows awakenings from REM (rapid eye movement) sleep, but dreams, as defined above, also occur during periods of non-REM sleep, sleep onset, and even periods of relaxed wakefulness for some subjects, leaving Foulkes (1985, p. 76) to conclude that "dreaming is a phenomenon which occurs in the presence of certain *mental* conditions: the relative absence of sensory stimulation and of the processing of such stimulation, the relinquishment of an Active-I, and persistent and relatively diffuse mnemonic activation."

Dreaming is thus a well-defined and distinct form of human thinking, yet a mode of thought that has been largely ignored by cognitive science. Fortunately, there are a few exceptions, notably Moser, Pfeifer, Schneider & von Zeppelin's (1983) computer simulation of a couple of dreams; and Mueller & Dyer's (1985) computational model of human daydreaming. Baylor & Deslauriers (1986-87; 1987-88), in their studies of morning recalls of night dreams, laid out a programmatic procedure for facilitating dream understanding by helping users recover aspects of their own dream formation process. They hypothesized that the planning component of dream generation can best be characterized as a script: a stereotyped sequence of actions in a more or less well-known situation (Schank & Abelson, 1977; revised by Schank, 1982). Scripts also contain information about the roles and goals of characters, about constraints on behavior, and about anticipated outcomes. In dreams, scripts appear as old patterns of behavior that instantiate current concerns in the dreamer's life and prescribe actions. In actual fact, of course, dreams deviate in major ways from scripts. Thus, it was hypothesized that deviations from script and the constraints they violate permit the dreamer to accomplish personal goals not prescribed by the script, which are often the major moments of meaning in dream understanding.

It is, of course, difficult to validate the notion that scripts are the best way of characterizing the planning component of dream construction. However, as a first step it would appear useful to know if there is agreement between the dreamer and independent judges as to what script, if any, may be underlying a dream. To this end an experiment was carried out to assess the degree of interjudge agreement. This is followed by the detailed analysis of one dream report where there was good agreement.

DESCRIPTION OF THE EXPERIMENT

Deslauriers for his doctoral research collected the laboratory dreams of seven volunteer male university students who spent two consecutive nights in the sleep

laboratory at Carleton University, Ottawa (Note 1). They had been briefed beforehand about the purpose of the experiment and were given instructions about scripts and how to recognize them, including three practice dreams followed by feedback and discussion with E.

To recognize the stages of sleep, three main electrophysiological signals were used: EEG, EOG, and EMG (Arkin, Antrobus & Ellman, 1978). Before retiring, electrodes were attached to the S's scalp (EEG), eyes (EOG) and chin (EMG). The signals were then amplified, sent to a digital converter and transferred to a video monitor. Following 8-12 minutes of each period of REM sleep, E awakened S and asked him, first, to report his experience prior to the awakening and to indicate the order in which the events occurred; second, to describe his feelings during the experience; third, to identify what well-known situation or script the dream reminded him of, if any, as well as to indicate the information in the dream he used to identify such a script; fourth, to describe what seemed unusual or bizarre in the dream; and, finally, to report any events from his daily life that the dream may have reminded him of. S then returned to sleep.

Transcripts of the verbatim dream reports were given to two independent judges (Note 2). They were required to perform the same task as the Ss, namely, to identify a script, if any, and to specify the criteria they used for identifying it. The judges were not, of course, informed of the Ss' script identifications.

RESULTS

For the seven Ss reported here, 48 REM dreams were reported, five of which were eliminated (two because at least two judges agreed that there was no script; and the other three because at least one judge identified more than one script) leaving 43 REM dreams. The dreamer's script identifications as well as the two judges' were then compared by pairs to evaluate interjudge agreement, representing 129 comparisons for the 43 dream reports. The comparisons were categorized on the following 5-point scale:

0 - Total disagreement: two totally different and unrelated scripts
1 - Loose thematic agreement: two different but only loosely related scripts (e.g., watching an overfed frog get smashed and trapping an animal)
2 - Thematic agreement: two different but related scripts (e.g., discussion in a restaurant and sitting in a restaurant)
3 - Semantic agreement: Essentially the same script but expressed in different words (e.g., making conversation and discussion with a friend)
4 - Complete agreement: the same script expressed in the same or nearly the same words.

The results appear in Table 1. Out of 43 dream reports, there was complete agreement by all three judges (counting the dreamer as a judge) on seven dreams (16.3%) and complete agreement by two of the three judges on 12 additional dreams (adding up to 44.2%). The dream scripts that all three judges agreed on are perhaps worth listing: a

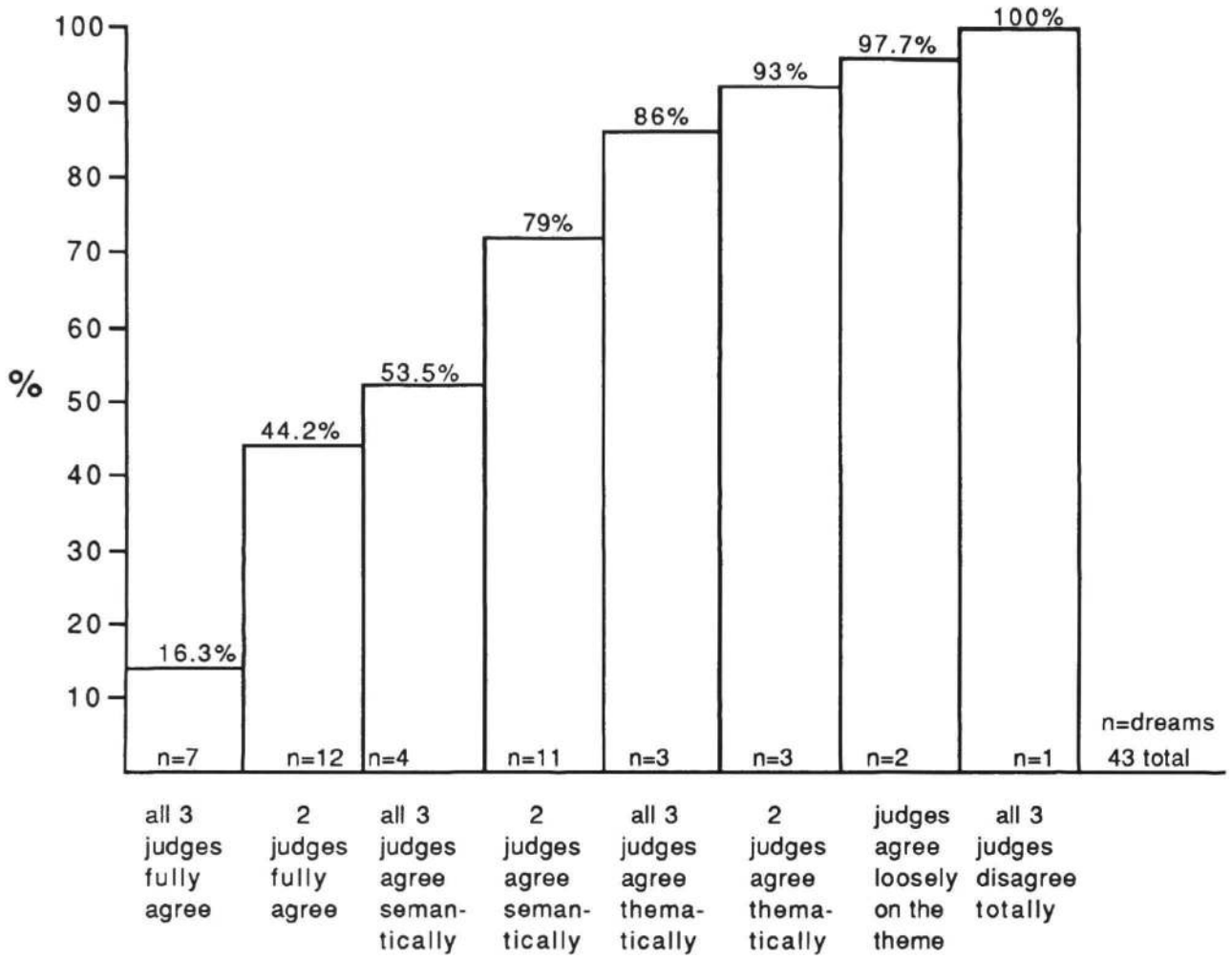


Table 1: Cumulative percentage of dreams by each category of agreement

basketball game, a football game, jogging in the park, fighting, being in a car, shopping with friends, argument with a roommate.

Relaxing the criterion to include the scripts that were rated semantically similar, there were four dreams where all three judges agreed on essentially the same script and another 11 where two of the three judges so agreed. All together this accounts for 34 of the 43 dreams, or 79.0%.

DISCUSSION

To us, 79% appears to be a surprisingly high number, indicating that judges can agree on a script underlying 7 or 8 REM dream reports out of 10. But just what does this interjudge agreement reflect? Would Ss be able to generate and agree on a core set of actions for each dream script, as in the Bower, Black & Turner (1979) experiment? Perhaps this is how they made their script identifications in the first place. Of course, when comparing the richness of the dream narrative with the scripts as identified, the

script virtually never explains all the events in a given dream report. Dreams are almost never limited to simply reproducing typical situations. Thus, it would seem then that the dream scripts, as identified, provide a framework or skeletal structure for generating the major dream scenes, but for the flesh of the dream narrative, we need to look more closely at one dream report in detail.

The following dream report, the second REM awakening of the night, was produced by a male S in his late twenties who, in addition to being a student, also worked as an airport security guard.

It started when I was talking to somebody, I guess a fellow worker. I was working some kind of patrol or Mountie or Mountain ranger type of guard. We were carrying these modified shotguns. I think it was a big power rifle with a full stock maybe. I was carrying the rifle with the stock open resting on my belt. It was out in the country, in the hills, it seemed to be a mountainous area. It was summertime; it was warm out.

I left him and I was walking across the highway on my way to Dunden. It was like six lanes of highway. The first two lanes came around the corner. I had to get across the highway quickly, the first two lanes anyhow. On the highway, on the two corners coming out, there was a truck that came around. There were quite a few people in it. There was a back seat and I remember three men in the back, one of them had a broken arm. As they went by, I continued walking.

I looked behind for traffic but there was no traffic, only this lady coming at me. I went to this building and this is when I met this lady, when I had just crossed the highway. She was carrying a rifle. She was bitching at me that I wasn't carrying mine properly. I said "there's nothing wrong with the way I'm carrying mine; there's no difference from yours." It seems to be the argument as to whether or not the barrel was above my head or anybody's head as we walked into the office. She kept on bitching. I ignored her.

We walked into the building where I worked, like a ranger station. There were three or four fellow employees. She kept staring at us. One employee was carrying a rifle. He figured he's get rid of her, so he took her rifle and stuck it in his mouth and was able to hold it just with his teeth by the end of the barrel as to express "it doesn't matter how you carry a damned rifle" or something. He was trying to shock her and to tell her to mind her own business. She didn't find it funny. I was standing to the right of her and I was surprised there were no comments on her part about what he had done. Somehow she handed me a pistol, she pointed it at me. I guess we kind of give her shit for that. Then I woke up.

When, after reporting the dream, E asked S if the dream reminded him of any events from his daily life, he said that he was "upset" about a teaching assistant, a woman he strongly disliked whom he had been talking about before going to sleep. He said "she is prejudiced towards men ... if somebody took a look at the marks she gave in the class, he would probably find out that men have lower than women." Apparently, in a recent meeting with the teaching assistant S had failed to have his mark changed, and he was obviously angry at her: "I think she's a real bitch."

The dreamer identified this script as what was for him "a day at work" where he, in fact, carried a shotgun. One judge called it "Patrol guards talking" and the other "On ranger

patrol." This is thus a script where there was semantic agreement among the three judges.

The script does appear to instantiate one of the dreamer's current concerns. The script of "a day at work" is for the dreamer a well-known situation where his competence is well established. The goal of a ranger patrol script is to protect a territory from intruders, and the dreamer, by the choice of this script, is attemping in a metaphoric problem space (Baylor & Deslauriers, 1986) to solve the problem of how to pass an academic course where he has to deal with a hostile teaching assistant. For example, the lady, who in the dream has the role of both intruder and critical superior, reminds him of the teaching assistant; this is the most obvious metaphoric referent. Also, in light of S's statement about how he feels the teaching assistant "is prejudiced towards men," crossing the highway without getting run down appears to convey metaphorically how the dreamer is trying to get through the course without ending up as a casuality like one of the men in the truck.

The script also provides the planning mechanism for generating a sequence of scenes, comparable to Mueller & Dyer's (1985) scenario generator. Specifically, the scenes prescribed by the script are (1) getting ready for patrol duty (corresponding to the first paragraph of the dream report); (2) walking the patrol (paragraphs two and three); and (3) going back to the station after patrol duty (paragraph four). This is the skeletal structure of the dream.

Within each scene, of course, there are many actions: some of them fall within the confines of a typical patrol script: for example, "talking to a fellow worker," "carrying a shotgun," leaving him and "walking across the highway," etc. Other activities, however, are not predicted by the script: for example, "a truck .. came around ... with three men in the back, one of them had a broken arm," "there was ... this lady coming at me...she was carrying a rifle," etc. While these actions are not prescribed by the script, they can be accommodated by (or assimilated to) it. This second class of actions was called tag actions, "causally related events not predicted by the script," in memory experiments by Reiser, Black & Abelson (1985, p. 92) and Graeser, Woll, Kowalski & Smith (1980).

Finally, the third class of actions are those that are so bizarre that they cannot reasonably be assimilated to the script: what Baylor & Deslauriers (1986-87; 1987-88) called the deviant actions. Indeed, they are judged deviant or bizarre exactly because of the difficulty of assimilating them to the current dream context. For example, "one employee ... took her rifle and stuck it in his mouth and was able to hold it just with his teeth by the end of the barrel." Taken together, these three classes of actions constitute the body of the dream narrative.

Underlying the dream narrative, of course, is a goal structure. For example, the top level goal appears to be to restore the dreamer's sense of self-esteem (ego consolidation, Dallett, 1973), which has been seriously compromised by S's confrontation with the female teaching assistant. The strategies (or control goals, in Mueller & Dyer's terminology) for re-establishing self-esteem are three: first, he tries to

prove to the woman rationally that her criticism is unfounded since "there's nothing wrong with the way I'm carrying mine [rifle]; there's no difference from yours." This produces no results, so, second, he tries to ignore her, but she is persistent and follows him to the ranger station. Since neither strategy has been effective, there is now a radical deviation from script: a second man, perhaps acting as a surrogate or "dissented" aspect of the dreamer (Piotrowski, 1986) "figured he'd get rid of her, so he took her rifle and stuck it in his mouth...." "He was trying to shock her and to tell her to mind her own business." She is unimpressed. Suddenly, however, her rifle becomes smaller when she hands the dreamer a pistol, which she points at him. While this is an aggressive act, still the dreamer has succeeded in disarming her. The goal of self-esteem has been realized and the dreamer is now in a position where he is no longer threatened by the woman. In fact, the men now even take the offensive and "give her shit for [pointing the gun at him]." Given the experimental design, we do not have enough information about S to know the nature of the personal goal that may be underlying this deviant action; it is, of course, tempting to postulate some sort of sexual motif, which would presumably violate the constraints of a ranger patrol script.

In conclusion, it might be useful to compare and contrast our analysis of night dreams with various aspects of Mueller & Dyer's (1985) DAYDREAMER model, which must serve as a touchstone for any subsequent theory of dream production. First, night dreams usually begin with a more elaborate re-presentation of the situational description, often, as in the above example, in a metaphoric problem space. Second, in DAYDREAMER, emotions activate control goals which results in daydreaming. Similarly, we found anger, probably also accompanied by humiliation, to activate the personal goal of restoring self-esteem, which results in this night dream. Third, in DAYDREAMER there are both control goals (rationalization, revenge, failure/success reversals, and preparation), which guide the scenario generator, and conflicting personal goals (health, food, sex, friendship, self-esteem, love, achievement, etc.). In our analysis, we found that the personal goal of self-esteem appears to function as a top level "control goal" and guides the scenario generator, whereas what Mueller & Dyer (1985) called control goals appear to function as strategies in the service of the personal goal of restoring self-esteem. Fourth, in DAYDREAMER "the basic mechanism for scenario generation is planning ... i.e., generating a sequence of actions necessary to achieve a goal" (p. 124). Similarly, in our analysis, scripts function as the skeletal structure of the planning mechanism. Night dreams do, of course, differ from daydreams in their hallucinatory quality, in their relative absence of sensory stimulation and its processing, and in the relinquishment of an Active-I, so an eventual a model of night dreaming should be discriminated from a model of daydreaming.

ACKNOWLEDGMENTS

Note 1. We are much indebted to Drs. Robert Hoffmann and Alan Moffitt for their instruction, support and permission to use their sleep laboratory, without which this experiment would probably not have been carried out.

Note 2. Many thanks to Mr. André Renaud and Dr. Howard Schachter who graciously served as judges in the painstaking analyses of the dream transcripts.

REFERENCES

Arkin, A.M., Antrobus, J.S., & Ellman, S.J. (1978). The mind in sleep. Hillsdale, NJ: Erlbaum.

Baylor, G. W. & Deslauriers, D. (1986). Understanding dreams: Method, maps and metaphor, Dreamworks, 5:1, 46-57.

Baylor, G. W. & Deslauriers, D. (1986-87). Dreams as problem solving: A method of study -- Part I: Background and theory, Imagination, Cognition and Personality, 6(2), 105-118.

Baylor, G. W. & Deslauriers, D. (1987-88). Dreams as problem solving: A method of study -- Part II: The oral defense dream, Imagination, Cognition and Personality, 7(1), 23-45.

Bower, G.H., Black, J.B., & Turner, T.J. (1979). Scripts in memory for text, Cog..Psych., 11, 177-220.

Dallett, J. (1973). Theories of dream function, Psychological Bulletin, 6, 408-416.

Foulkes, D. (1985). Dreaming: A cognitive-psychological analysis. Hillsdale, NJ: Erlbaum.

Graeser, A. C., Woll, S. B., Kowalski, D. J. & Smith, D. A. (1980). Memory for typical and atypical actions in scripted activities, J. Exp. Psychology: Human Leaning and Memory, 6:5, 503-515.

Moser, U., Pfeifer, R., Schneider, W., & von Zeppelin, E., (1983). Experiences with computer simulation of dream processes. In W. P. Koella (Ed.), Sleep 1982 (pp. 30-44), Basel: Karger.

Mueller, E.T. & Dyer, M.B. (1985). Towards a computational theory of human daydreaming, Proc. 7th Conf. Cog. Sci. Soc., 120-129.

Piotrowski, A. A. (1985). Dreams: A key to self-knowledge. Hillsdale, NJ: Erlbaum.

Reiser, B. J., Black, J. B. & Abelson, R. P. (1986). Knowledge structures in the organization and retrieval of autobiographical memories, Cog. Psych., 17, 89-137.

Schank, R. C. (1982). Dynamic memory. New York: Cambridge Univ. Press.

Schank, R.C. & Abelson, R.P. (1977). Scripts, plans, goals, and understanding. Hillsdale, NJ: Erlbaum.

THE RIGHT OF FREE ASSOCIATION:
Relative-Position Encoding for Connectionist Data Structures

John A. Barnden

Computing Research Laboratory
New Mexico State University

DATA STRUCTURING IN CONNECTIONIST SYSTEMS

The challenge presented to connectionism by high-level cognitive processing — which includes reasoning, planning, and some aspects of natural language understanding — is gaining increasing recognition, and attempts to meet it are becoming more common. The main technical difficulties are listed in Barnden (1983, 1984, 1986, 1988a), McDermott (1986) and Norman (1986). They include the well-known variable-binding problem and the problem of accounting for complex, temporary, novel data structures. An example of such a data structure is an internal rendering of the information in the sentence "John believes that Pat gets angry whenever Tom talks about going to Tibet".

The thesis of this paper is that a promising way to approach the mentioned challenge is to encode complex temporary data structures by means of two somewhat atypical techniques, which I call "Relative-Position Encoding"(RPE) and "Pattern-Similarity Association" (PSA). These are answers to the following question, which lies at the heart of the technical difficulties mentioned above:

The Temporary-Association Question
How are pieces of information put into temporary association with each other in a connectionist system? (For instance, how are the different parts of a complex temporary proposition put together? How are variables bound to values?)

A variety of connectionist techniques have been proposed in answer to this. It is hard to categorize them satisfactorily, but the following rough classification will suffice for present purposes:

- *weight-change techniques*
- *binder techniques*
- *pattern-relationship techniques* (including PSA)
- *positional techniques* (including RPE).

These classes should be thought of as possibly-overlapping regions within a complex space of possibilities, rather than as being rigidly delineated.

Weight-change techniques involve temporary weight changes on connection paths between nodes or subnetworks representing the information items to be put into association. An (excessively) simple example would be a temporary weight increase on a single connection between two network nodes, of which one represents the idea of being hungry and the other a particular person, John: the intent being to temporarily represent the proposition that John is hungry. More complex examples would involve weight changes on groups of connections or connection chains joining large and possibly distributed sets of nodes. Important recent examples of weight-change techniques are provided by systems using multiplicative connections [Pollack 1987] or programmable networks [McClelland 1986].

The simpler forms of binder technique involve a temporary activation change on a "binder" node dedicated to the particular combination of nodes or subnetworks representing the information items to be put into association. More complex forms of binder technique allow the binder to be a subnetwork rather than a single node, and/or allow different activation states of the binder to indicate different bindings. Smolensky (1987) has provided a general framework encompassing binders. Binders appear in the systems of Cottrell (1985), Derthick (1987), Touretzky & Hinton (1985), and others.

The binder is usually connected in a straightforward way to the nodes/subnetworks it can bind, and in that case an unusual activation level on the binder can be construed as a temporary marking of a connection chain joining those nodes (cf. Feldman's (1982) dynamic connections). The binder technique is

thereby closely related to weight-change techniques.

In pattern-relationship techniques, the temporary association consists of the exploitation of some given relation on activation patterns. An example is provided by the "reduced descriptions" technique [Hinton 1987, Touretzky & Geva 1987]. However, what is more relevant to the present paper is a special type of pattern-relationship technique, called "*Pattern-Similarity Association (PSA)*" techniques. A (over)simple example of the idea is as follows. Imagine two subnetworks h and j, each divided into two parts called the Information Part and the Associator Part. The temporary proposition that John is hungry could be encoded by letting the Information Parts of j and h contain activity patterns representing John and the idea of being hungry, respectively, and letting the Associator Parts of h and j contain the same activity pattern X. The subnetworks h and j are deemed to be temporarily associated by virtue of this equality of their "associator" patterns (X). In more sophisticated versions of the idea the required similarity of patterns need not be equality, and, in principle at least, there need be no separate Associator subnetworks — associator patterns could be mixed in with information patterns.

There is a sense in which PSA techniques are really binder techniques of an advanced sort. In the simple example above, if several Associator Parts contain the same activity pattern X, then the union of these subnetworks can be construed as a binder containing a specific activation pattern indicating the binding together of the subnetworks. The binders thus defined, being unions of Associator Parts, are highly distributed, overlapping subnetworks. They are so different from the binders usually entertained in connectionist models that it is worth isolating PSA techniques as a special class.

Limited forms of positional technique crop up in many connectionist systems, especially those concerned with visual perception. The idea is best approached by means of a simple prototypical example. Consider a word-perception system that has a separate register-like subnetwork for each letter position in words. A particular pattern of activation distributed over the ith subnetwork would represent the presence of a particular letter at position i. Implicitly, therefore, the patterns of activation in the subnetworks encode *temporary associations* — relative positions — of letters in a word. Thus, the technique is termed "positional" because temporary association is achieved *merely by placing suitable activation patterns, encoding the information items to be associated, in suitable "positions" in the total network.* There is no weight change, activation of binders, or use of special associator patterns. (The name of the positional technique does *not* derive from the fact that what is encod*ed* in many applications is position in words or some other type of space.)

The (quasi-)connectionist sentence-parsing system of Charniak & Santos (1987) uses a more interesting and advanced positional technique that is highly germane to this paper. The model is centered on a two dimensional array of registers, which could in principle be implemented as connectionist subnetworks. By putting the registers into suitable states representing syntactical categories, the array can hold parse trees. The tree structure is encoded to a significant extent by the relative positioning of states in registers. We therefore call the technique a form of "*Relative-Position Encoding (RPE)*".

The working hypothesis that, to address the abovementioned "challenge" adequately, what is needed is an advanced form of RPE, has underlain my own development of connectionist models [Barnden 1985, 1986, 1987, 1988a,b], which has proceeded independently from the work of Charniak and Santos. The next section will sketch my present model, called "Conposit", which relies on an RPE technique broadly similar to that of Charniak & Santos, but more general and thorough-going than theirs. Conposit makes crucial use, too, of a PSA technique.

SKETCH OF CONPOSIT

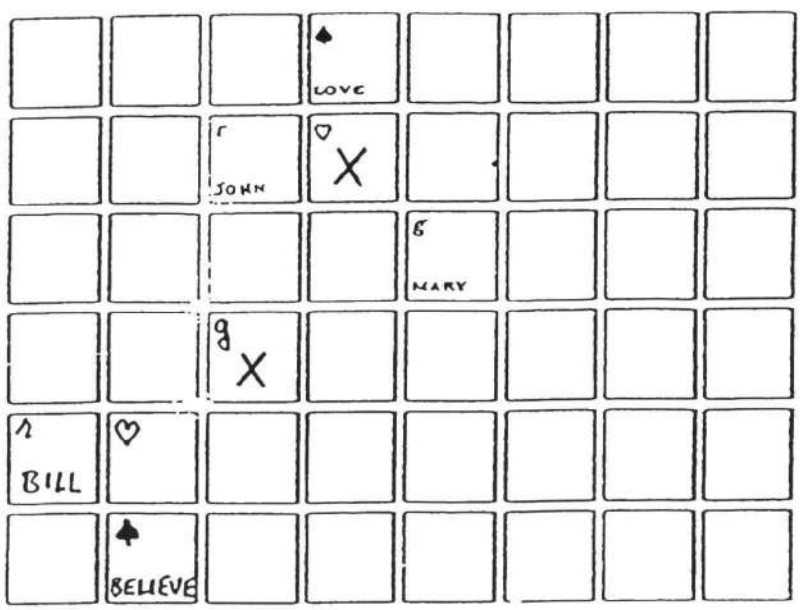
This section is of necessity highly simplified, and concentrates on Conposit's representational tools rather than its processing methods. Representation and processing details can be found in Barnden (1986) supplemented by Barnden (1987, 1988a,b).

Conposit is currently only "quasi-connectionist", in the sense of being a computational architecture whose components can be straightforwardly implemented in connectionist terms [see Barnden 1986 for some suggestions]. A detailed connectionist implementation will be simulated in future research. The two-stage strategy adopted — mapping information processing to an intermediate-level model, and later mapping this to the detailed connectionist level — is done in a spirit analogous to good design practice in computer science. The intermediate level has all along been constrained by the need for later connectionist implementation.

The versions of Conposit simulated to date are centered on a 32×32 array of registers, called the "configuration matrix (CM)". (The small, 2D nature of the CM is motivated by the fact that Conposit is intended to be a rough, preliminary model of high-level processing in brain cortex, and the CM is taken to be implemented as a small region of the cortical sheet. Individual registers are taken to be implemented as cortical columns.) The purpose of the CM is to hold *temporary* data structures being manipulated in reasoning and other high-level cognitive tasks. Conposit as currently simulated has no long-term memory for data structures, but see Barnden (1986) for an LTM proposal.

The content of the CM at any moment is given by a function σ from the CM registers to a state set B. Each member of B is an ordered pair (s, h), where s is a "symbol" and h is a vector of ON/OFF values for a fixed set of "highlighting flags". Both s and h for a given register in the CM are assumed to be implemented as activation patterns over part of the small neural/connectionist subnetwork forming the implementation of the register. Any symbol can be placed in any register. A symbol may have a specific representational function, such as denoting a particular person or a particular type of relationship among people. The significance of highlighting will become apparent.

The associations between the parts of a temporary data structure are encoded by (a) *adjacency relationships among values in CM registers,* often supplemented by (b) *sharing of symbols by several registers.* The RPE and PSA aspects of Conposit reside in (a) and (b) respectively. The encoding is exemplified by the Figure below, which shows a possible state of a region within the CM.



This region contains a representation of the proposition that *Bill believes that John loves Mary.* Each square shows a register. The words and capital letter X indiacte specific symbols s. Absence of symbol word indicates an occurrence of a special "null" symbol. The spade, heart, 'r' and 'g' symbols indicate ONness of four specific highlighting flags, called black, white, red and green respectively. The

symbol LOVE permanently denotes the class of all possible situations in which one person loves another. Any register containing this symbol also temporarily denotes that class. Any white-highlighted register adjacent to a LOVE-containing, black-highlighted register is deemed to denote, temporarily, a *member* of the class — that is, a loving situation. Such a register appears near the middle of the Figure. Any red-highlighted register adjacent to a register denoting a loving situation is deemed to denote the agent of the loving; similarly for green highlighting and the object of the loving. Thus, the upper "subconfiguration" of register states shown in the Figure encodes the proposition that John loves Mary.

So far, then, temporary association is a matter of exploiting the permanent, 2D structure of the CM, by having symbol occurrences and highlighting in suitable relative positions. In fact, the encoding relies only on the undirected *adjacency* relationships in the CM. Ways of using the CM that make broader appeal to the permanent structure in the CM — notably by using the CM as a map of some region of space in certain simple types of spatial reasoning — are discussed elsewhere [Barnden 1986, 1987].

The symbol X in the Figure is one of a special class of "unassigned" symbols, that do not permanenty denote anything and are akin to variables in a logic. However, by virtue of its position in the head register (i.e. the white-highlighted one) in the John-loves-Mary subconfiguration in the Figure, X is deemed to *temporarily* denote the loving situation denoted by that register. But X also appears in another register in the Figure, and makes that register also denote the loving situation. Thus, symbol-sharing is being used to temporarily associate registers in the sense of making them temporarily denote the same thing. This is an instance of Conposit's use of PSA (Pattern-Similarity Association). Because the lower X-containing register in the Figure denotes the situation of John loving Mary, the lower subconfiguration in the Figure encodes the top level of the proposition that Bill believes that John loves Mary. (PSA can also be used to split up propositions like John-loves-Mary into several pieces.)

The processing of the short-term data structures in the CM is performed by internal and external "circuitry" — system components that will be mapped straightforwardly into a connectionist implementation. The internal circuitry mediates mainly neighbor-neighbor interaction among registers in the CM, whereas the external circuitry embodies hardwired *condition-action processing rules*. Rules can detect particular configurations of symbols and highlighting states in the CM, by means of highly parallel detection circuitry that involves further two-dimensional register arrays called *location matrices* (see below). The response of a rule is to send a complex sequence of "CM signals" to the CM. The generation of the sequence can involve conditionals (testing the CM state), loops, and a simple form of non-recursive routine calling. A CM signal affects the CM in a highly SIMD-like, register-local, parallel fashion: the signal is distributed identically to each CM register, whereupon different registers change state differently, according to their own current states and those of their immediate neighbors.

A CM signal can have one of a number of effects, such as: changing the states of some highlighting flags in each register that is highlighted in some specified way and that has at least one neighbor highlighted in some other specified way. It is also possible for a signal only to have an effect on a randomly chosen register satisfying the highlighting conditions, rather than on each such register. Barnden [1986] details how the signals can be used to process data structures, and, in particular, to find free space for, and then create, new data structures in CMs. Specific information processing tasks are reported on in the next section.

Although there is no space here for a detailed account, there is a sense in which the binder technique for temporary association is used behind the scenes. The binders are the elements of some of the "location matrices"(LMs) alluded to earlier. An LM is a 32×32 matrix of registers, each containing an ON or OFF value. Suppose for instance that the system contains a hardwired rule triggered by the presence in the CM of a loving situation. Then the system must contain a certain LM with the following property: *an ON value at any position (x,y) in the LM indicates that register (x,y) of the CM currently denotes a loving situation*. More precisely, such a CM register is white highlighted and has a black-

highlighted neighbor containing the LOVE symbol (recall the Figure). To cut a long story short, the (x,y) register in the LM receives a connection path from the CM's (x,y) register and each neighboring CM register. As a result, ONness at (x,y) in the LM can be viewed as indicating the temporary association of the CM (x,y) register and its black-highlighted neighbor. Therefore, the LM (x,y) register, which is implemented as a connectionist subnetwork, is acting as a binder.

But it is essential to realize that Conposit's RPE is independent of such binders, in the sense that they serve *only* the subconfiguration-*detection* needs of hardwired rules: they are not brought in by the sheer requirement of *representation*. This is underscored by two observations. (1) Rules can analyze data structures by traversing them, through the use of highlighting movements in the CM, without having first detected them by means of the binders in LMs. (2) Later versions of Conposit will contain secondary CMs used for short-term storage purposes only, and having no attached LMs.

On the other hand, Conposit's PSA can be construed as being based on binders — of a highly atypical nature — in the way described in the first section.

SIMULATIONS OF CONPOSIT

The simulations (performed on NASA's Massively Parallel Processor) incorporate timing delays based on crude but reasonable neural implementation assumptions [Barnden 1986]. E.g.: 2 milliseconds is allowed for each logical gating operation in the neural "local circuits" hypothesized to exist in CM registers [Barnden 1986]; speeds of 1 and 10 meters/second are assumed for short and long neural signal travel respectively; and hardwired rule circuitry is generously assumed to be 5cm away from the CM. Therefore, there is a significant transmission delay (5 milliseconds) over long distances.

Barnden (1988a) reports experiments with two versions of Conposit. The first incorporates production rules for commonsense reasoning, one of which can be paraphrased as: *if a person X loves a person Y who loves a person Z (different from X), then X is jealous of Z.* Such rules exercise Conposit's handling of variable bindings. The rule executes in 515 milliseconds of simulation time. The rule deals with any situation satisfying the rule condition (and set up as the initial state of the CM), without massive duplication of circuitry for different bindings of the variables.

The second Conposit version in Barnden (1988a) converts a simplified form of parse tree for an active sentence into the parse tree for the corresponding passive sentence, in 1063 milliseconds of simulation time. The conversion parallels a task performed by the connectionist production system of Touretzky (1986). The initial tree is given as the initial state of the CM, and Conposit replaces it by the new tree. The rule has no massive duplication of circuitry for different choices of word in the sentence.

The Conposit version described in Barnden (1988b) engages in syllogistic reasoning, by embodying some core aspects of the Johnson-Laird's "mental model" theory [see e.g. Johnson-Laird & Bara 1984]. (Conposit could also accommodate more conventional logical processing of syllogisms.) An example syllogism is: *"Some athletes are beekeepers; all beekeepers are chemists; therefore, some athletes are chemists"*. This is represented in the CM by propositional subconfigurations analogous to those in the Figure above. Conposit's syllogism-processing rules (which involve no massive duplication of circuitry for different choices of object categories in syllogisms) use the syllogism premises to construct a random Johnson-Laird mental model — a sort of example situation involving several athelete, beekeeper and chemist "tokens". A rule then checks whether the syllogism conlcusion is consistent with the mental model. Conposit can run through a syllogism (which involves 6 rule firings, and the creation of 15 tokens on average) in about 2.5 seconds of simulation time. This appears to be fast enough for psychological plausibility, judging by the times allowed to human subjects in Johnson-Laird's experiments.

CONCLUSION

In response to the "challenge" mentioned in the first section, McDermott (1986) has advocated systems that contain symbols — reproducible patterns of network-node activity that can have distinct simultaneous occurrences. Conposit is based on just such symbols. Their use in a Relative-Position Encoding technique, combined with Pattern-Similarity Association, leads to a (quasi-)connectionist system capable of complex high-level symbolic information processing.

The temporary-association methods used by Conposit are particular points in a rich space of techniques for the analysis of which a general unifying framework would be beneficial. The following comments explain the intuitive basis of a general framework that I am currently devising, to accommodate the classes of technique discussed in the first section, and thereby to illuminate their relationships. In particular, the relationship of binder techniques and PSA techniques needs further clarification.

The various techniques discussed embody ways of *TEMPORARILY exploiting PERMANENT structure to achieve TEMPORARY association*. In the weight-change case the permanent structure is some set of connection chains in the total network, and the association is achieved by temporarily exploiting connection chains in the sense of putting them into particular temporary states. The binder case is similar, but the exploitation consists of changing the activation of intermediate nodes in connection chains. In the pattern-relationship case, at least in the special case of PSA, the permanent structure exploited is the appropriate type(s) of similarity among activation patterns, and the exploitation is the placing of similar patterns in suitable network regions. In the positional case, the permanent structure exploited is the way some register-like subnetworks are arranged in the total system architecture, and the exploitation consists of the placing of suitable activation patterns in the registers.

It is illuminating to apply this unifying view of temporary association (as temporary exploitation of permanent structure) to the way it is achieved in primary storage in ordinary computers. The associations are achieved in temporary states of the store that exploit various types of permanent structure. The *sequential allocation* technique for data structuring, where for instance the items in an array are placed in sequence in some set of consecutive locations, exploits the total order on the set of storage locations. The *pointer* or *linking* technique for achieving data structuring exploits the (partial) function from bit-strings, construed as addresses, to locations. A simple form of *content addressing*, in which two locations containing the same (sub)string of bits may be considered to be currently associated, exploits similarity relationships among bit-strings. In sum, sequential allocation, content addressing and linking temporarily exploit permanent structure defined over the set of locations, the set of possible bit-strings, and the mapping from bit-strings to locations, respectively.

It should be obvious how this view of computers can be modified to cover RPE and PSA in Conposit, taking the CM to play the role of the store and ignoring the linking technique. I am hoping to extend the view to cover positional and pattern-relationship techniques more generally, and to encompass also the binder and weight-change classes of technique. Particular tasks to be faced in constructing such a general framework will be to relate it to certain other general frameworks, such as Smolensky's (1987) account of roles, fillers, and variables, and Walters' (1987) account of connectionist information encoding.

ACKNOWLEDGMENT

I am indebted to NASA for unlimited free access to the Massively Parallel Processor at the Goddard Space Flight Center, through my membership in the MPP Working Group.

REFERENCES

Barnden, J.A. On association techniques in neural representation schemes. *Procs. 5th Conf. of the Cognitive Science Society*, Rochester, NY, 1983.

Barnden, J.A. On short-term information processing in connectionist theories. *Cognition and Brain Theory, 7* (1), 1984.

Barnden, J.A. Diagrammatic short-term information processing by neural mechanisms. *Cognition and Brain Theory, 7* (3&4), 1985.

Barnden, J.A. Complex cognitive information-processing: a computational architecture with a connectionist implementation. Tech. Rep. 211, Computer Science Dept., Indiana University, Bloomington, IN. December 1986.

Barnden, J.A. Simulation of an array-based neural net model. In *Proceedings of the First Symposium on the Frontiers of Massively Parallel Scientific Computation*. NASA Conference Publication 2478. 1987.

Barnden, J.A. High-level reasoning in a quasi-connectionist register-array model. Submitted to European Conference on Artificial Intelligence, 1988a.

Barnden, J.A. Commonsense syllogistic reasoning in CONPOSIT, a quasi-connectionist register-array model. Submitted to National Conference on Artificial Intelligence, 1988b.

Charniak, E. & Santos, E. A connectionist context-free parser which is not context-free, but then it is not really connectionist either. *Procs. 9th Annual Conf. of the Cognitive Science Society,* 1987.

Cottrell, G.W. Connectionist parsing. *Procs. 7th Annual Conf. of the Cognitive Science Society,* 1985.

Derthick, M. A connectionist architecture for representing and reasoning about structured knowledge. *Procs. 9th Annual Conf. of the Cognitive Science Society,* 1987.

Feldman, J.A. Dynamic connections in neural networks. *Biological Cybernetics, 46,*, 27-39, 1982.

Hinton, G.E. Representing part-whole hierarchies in connectionist networks. Manuscript, Computer Science Department, Carnegie Mellon University, Pittsburg, Penn., 1987.

Johnson-Laird, P.N. & Bara, B.G. Syllogistic inference. *Cognition, 16* (1), pp.1-61, 1984.

McClelland, J.L. The programmable blackboard model of reading. In J.L. McClelland, D.E. Rumelhart and the PDP Research Group, *Parallel Distributed Processing, Vol. II.* MIT Press, Cambridge, Mass. 1986.

McDermott, D. What AI needs from connectionism. Appendix to J.L. McClelland, J.A. Feldman, G. Bower & D. McDermott, "Connectionist models and cognitive science: goals, directions and implications," a report on an NSF workshop, 1986.

Norman, D.A. Refelections on cognition and parallel distributed processing. In J.L. McClelland, D.E. Rumelhart and the PDP Research Group (Eds.), *Parallel Distributed Processing Vol. 2,* Cambridge, Mass.: MIT Press, 1986.

Pollack, J.P. Cascaded back-propagation on dynamic connectionist networks. *Procs. 9th Annual Conf. of the Cognitive Science Society,* 1987.

Smolensky, P. On variable binding and the representation of symbolic structures in connectionist systems. Tech. Rep. CU-CS-355-87, Dept. of Computer Science, Univ. of Colorado, Boulder, 1987.

Touretzky, D.S. Representing and transforming recursive objects in a neural network, or "Trees *Do* Grow on Boltzmann Machines". *Procs. IEEE Conf. on Systems, Man and Cybernetics,* 1986.

Touretzky, D.S. & Geva, S. A distributed connectionist representation for concept structures. *Procs. 9th Annual Conf. of the Cognitive Science Society,* 1987.

Touretzky, D.S. & Hinton, G.E. Symbols among the neurons: details of a connectionist inference architecture. *Procs. 9th Int. Joint Conf. on Artificial Intelligence,* 1985.

Walters, D. Properties of connectionist variable representations. *Procs. 9th Annual Conf. of the Cognitive Science Society,* 1987.

Unsupervised Learning of Correlational Structure *

Andrew Chalnick
Computer and Information Science
University of Pennsylvania

Dorrit Billman
Department of Psychology
University of Pennsylvania

May 23, 1988

Abstract

People can learn simply from observation, without explicit feedback. Natural language acquisition is perhaps the most spectacular example, but unsupervised learning occurs in many domains. We present 1) a task analysis of a broad class of unsupervised learning problems and 2) an initial simulation based on the task analysis which successfully learns all the rule types identified in the analysis. Our task analysis characterizes systems of interpredictive correlational rules which could be the basis for category formation in unsupervised learning. For example, observation of various animals could lead to abstracting covariation rules among wings, feathers, and flight, and also among fins, scales, and swimming. These rules in turn could form the basis for the categories *bird* and *fish*. Our analysis identifies three types of predictive features and three types of rules which may be available in input: *universal*, *contrastive*, and *exception-based rules*. This analysis guided design of our learning procedures. Our simulation succeeds in learning all three rule types. This is difficult because procedures which facilitate learning one rule type may inhibit learning another. Further, our simulation is restricted in psychologically motivated ways and succeeds despite these requirements. We know of no other simulation or modeling project which addresses exactly this class of learning problems. Our results demonstrate the existence of successful procedures. However, we believe our most valuable contributions are our task analysis and framework for testing the power and limits of domain-general learning procedures applied to unsupervised learning problems.

Introduction

What procedures allow learners to discover categories from observation of instances? When contrastive feedback is provided people (and computers) can use this to guide learning. When discriminative feedback about category membership is not provided, a greater burden is placed on the learner to discover structure in input and set up sensible categories. Our project investigates what learning procedures are sufficient for successful learning under these conditions. One goal of the project is to develop a successful learning program; the more fundamental goal is to develop a model of the learning task which will allow us to investigate the effects of component learning procedures on different aspects of the learning problem.

Our core principle for explaining observational learning is internally generated feedback: learners compare predicted and observed properties and use the match or mismatch to guide learning. When input provides interpredictive relations among feature values, this structure can be discovered using internal feedback. Prior work began exploration of learning under these constraints [Billm87b]. This paper presents a more sophisticated model.

Systems of correlations are central to learning categories from observation. Furthermore, representation of correlational structure is intimately linked to category use: The primary purpose of categories is to organize knowledge to allow sensible inferences. Given enough information to classify something as a bird, this licenses additional inferences: for example, it hatched from an egg and will lay eggs if female. If we observe a new property for some particular bird, say, eating worms, we may generalize that property to other members of the same category. While these inferences are certainly not correct all the time, they provide us with a valuable way of extending our knowledge. Rosch [Rosch78] has pointed to the importance of correlations in defining category structure. She argued, however, that correlations are important in leading a cultural group to discover or rely on a category, but not that individuals use this correlational structure in category learning. We are interested in how an individual might use discov-

*We thank Richard Billington and Lyle Ungar for valuable input and discussion about this research.

ery of correlations as the basis for category learning. Prior work has found that individuals do use correlational structure in learning categories [Garne74, Billm87b, Billm87c].

In unsupervised learning, correlational structure is even more important than when categories are designated for the learner and explicit feedback about membership is provided. In learning to distinguish categories A and B with explicit feedback, finding any attribute which predicts category membership is sufficient. The learner has no need to notice whether various predictor features covary with one another. When no feedback is specified externally, concept learning will be largely driven by discovery of interpredictive relations among feature values. That is, subject's may discover coherent patterns in the observed examples and use this to set up categories.

We can understand the learning problem better by analyzing the structure which is potentially available in input. Specifically, we can identify three types of predictive rules and three classes of features. The three types of predictive rules are universal, contrastive, and exception-based. *Universal* rules apply to all instances in the domain; if the domain of learning is animals, then the system should discover that all animals eat and breathe. *Contrastive* rules could be used to divide the domain into major, contrasting classes. In learning about a wide range of animals, we want the system to discover that fins, scales, and swimming all go together, as do wings, feathers, and flying. Discovery of these rules was the initial goal in designing the system. Clusters of such rules can then form the basis for contrasting categories such as fish and birds. We would like the system to learn these regularities even when exceptions occur. Finally, *exception-based* rules represent information about the exceptions themselves; if an animal has wings, but looks like it is dressed in a tuxedo, it will not fly. The system should be able to learn about bats, whales, and penguins.

Procedures which facilitate discovery of one class of these rules may often inhibit learning rules of another class. Learning universal rules requires good sensitivity to very general patterns. Learning rules about exceptions requires good sensitivity to quite specific patterns. Learning rules which would form the basis of maximally coherent categories requires good sensitivity to features which are related to many other features. Thus designing a system capable of learning all three classes requires accommodating conflicting demands. This problem has not been highlighted in prior work, because the issue is much less important in learning with feedback. Specifically, there is no pressure to learn universal properties when learning categories from contrastive feedback. However, when a system is designed to seek good predictions, it must be prevented from focusing too much on vacuous predictions of universal properties.

Parallel to the three types of rules are three types of features. *Universal* features do not vary across the entire set of objects in the problem domain. *Contrasting* features are those which both vary across the domain and covary with other features. These are the features which we intuitively think of a defining separate classes. Features such as mode of locomotion, type of body covering, type of limb, and distinctive location or habitat divide up major classes of animals. *Idiosyncratic* features are highly variable across the domain, but do not generally covary with other features. Their predictive value is primarily in conjunction with several other features, to identify an individual or exception. Learning about penguins may require attention to distinctive coloring, even though color may not be widely predictive. Our learning model uses this analysis of universal, contrasting, and idiosyncratic features in the procedures which control learning.

Problem Definition

Our modeling begins with three fundamental assumptions about learning correlational patterns. First, people and other cognitive systems do learn about patterns of feature correlations, even in unsupervised learning. Second, this information is represented directly and locally, as in classifier [Holla75, Holla86] and production [AKB79] systems, not indirectly in a set of weights distributed across a system [Rumel86]. Direct representation of rules allow other mental procedures, as in inference and transfer, to selectively operate on representations of different regularities. Third, all three types of rules, universal, contrasting, and exception-based are important components of learning about the correlational structure in input. In addition, we place several constraints on the available information and resources.

1. The information available to the learner is limited. No feedback is provided and learning takes place from unsupervised observation of examples. Much natural learning is informal and untutored. Feedback is often sporadic,

unreliable, or unavailable. By modeling learning with no designated feedback we investigate the most difficult case; models for learning with feedback can be set up as an easier, special case [Billm87c]. Most researchers have addressed learning with feedback. Whether the learning criteria is predicting category membership [AKB79] or earning as much of a target resource as possible [Holla86], it is directly specified for the learner. In our task, the learner must discover which features are predictable as well as which features are predictive.

2. Memory, either storage or retrieval, is limited. Some specific information may be preserved. However, we do not allow learning procedures which operate over the set of previously seen objects. Rather, an observation affects the learner's state of knowledge but no representation of the object as an individual need be retained.

3. The learner's initial knowledge is limited. Learning should not depend heavily on the initial state of the learner. First, the learning procedure should be sufficiently robust so that the order of presentation of examples does not profoundly change the course of learning. Second, learning should not depend on extensive prior knowledge of the domain. We are interested in specifying general learning procedures which can apply even where the learner lacks much knowledge initially.

4. In general, we want our learning procedures to apply homogeneously across rules without reference to rule content. This approach contrasts with those where the strength of the learning procedures depends on the content of old knowledge.

There are undoubtedly many circumstances where the conditions of learning are not so austere. We are interested in investigating this difficult class of learning problems because we believe that these circumstances will tell us most about the strengths and limits of general, data-driven learning procedures. We investigate learning of systems of structured representations, given minimal initial knowledge and minimal information in input. Our simulation operates under these constraints.

Representation

The representation format is described here. This, together with the procedure description to follow, specifies all the apriori information built into the system. The learning model is independent of the particular domain. However, for a learner to succeed, input must provide at least some contrastive or universal features and the learner must represent these. For this implementation, the domain consists of birds, fish, mammals, whales, penguins and bats. Input objects are represented as vectors of feature values, where all features must be specified. Thus a typical object might be a red, furry land animal with legs which weighs 200 lbs., and eats and breathes.

Representation of Features

Both observed objects and internal rules are represented in terms of the same set of features. This feature set is not currently altered by learning. The simulation runs reported here used seven features: breathes (t or f), eats (t or f), color (blue, yellow, red, white, black, green, tuxedo, brown, pink), weight (coded into 7 distinct ranges), locomotion mechanism (legs, wings, fins), habitat (land, air, water), and body covering (hair, feathers, scales). Each feature has two associated parameters, salience and variability. Feature *salience* is a function of predictive success across all the rules in which the feature participates as a predictor. The feature *variability* is estimated from storing the set of recently observed values of that feature. This requires maintaining minimal information about the distributions of feature values. Contrastive features are variable and salient. Universal features are homogeneous. Idiosyncratic features are variable and have low salience. This information about features is used in the learning procedure.

Representation of Rules

Knowledge of regularities is represented in conditional rules. Each conditional rule consists of a condition and an implication. The condition specifies the values of a proper subset of the observable features. The implication specifies the value of one predicted feature. A conditional rule might specify that, if something has scales and fins, then it travels around in the water. Each conditional rule has an associated strength estimate. Rule strength is a function of the rule's

predictive validity, the salience of the features in its condition, and the variablity of its predicted feature. The present work differs from an earlier project [Billm87] in that we now allow multiple features in the condition. This is a fundamental change. First, when only single feature conditions are specified, it is feasible to enumerate all the representable rules. Then the learning procedure need only select good rules from among an initially instantiated set. When multiple features are allowed, the combinatorial explosion means that the learner must not only decide which of a number of rules is the best, but it must also generate these rules. Second, it changes the representational power of the system. Learning higher order regularities, subpatterns, and exceptions requires use of multiple features.

Procedure

The learning procedure consists of major and minor cycles. In the minor cycle, objects are presented to the learner and tested. The major cycle removes weak rules and replaces them with new, potentially stronger rules. The rules compete with one another, with their level of success based on their ability to explain the domain. The rules which provide the best model will survive while the others will be removed.

The Minor Cycle

The learner first picks a random object to examine, samples a set of features from the object, and then picks a rule to test given the sampled feature set. *Focused sampling* alters selection probability of the features sampled in observing an object. It directs attention to features which have proved predictive in the past. When a feature is sampled, its variability is updated.

With multiple features in the condition, it is unlikely that a particular set of feature values will find an exact match in the condition of some rule. Thus, we need a partial match value. The value is a function of 1) the number of conflicts between the rule and the set of sampled features, 2) the number of matches, and 3) the variability of the action feature of the rule (this helps avoid making vacuous predictions). A rule is then picked probabilistically as a function of its match score.

After a single rule is picked it can either be tested or generalized. A rule is generalized if some part of the rule's condition conflicts with some feature values of the object. To generalize, we remove the conflicts and add the generalized rule back into the rule base. The generalized rule is now free to compete with the parent rule. Only a limited expansion of the rule base is allowed in one major cycle. If a generalized rule is formed, its prediction is tested; if not, the orginal rule is tested. If the prediction is correct, the rule's predictive validity and the salience of the features in the condition are increased. If unsuccessful, they are decreased. The modification is done in accord with a *delta* rule. The delta rule revises a parameter by moving its current value a certain percentage closer to minimum or maximum value, depending on whether the rule or feature is being increased or decreased.

The Major Cycle

A major cycle follows a block of many minor cycles. Weak rules are dropped from the rule base and new ones inserted for testing. New rules are generated by sampling random objects and creating rules with conditions containing values for all but one of the features with the remaining feature in the implication part of the rule. Thus the new rules are maximally specific. The size of the rule base is expanded by the number of strong rules found in the prior block of learning cycles. This expansion allows new rules to compete more successfully. Finally, each rule's predictive validity is decremented a fixed percentage. This tax helps weaken and eliminate irrelevant rules.

Simulation Evaluation

The primary goal for the initial development phase of the system was a sufficiency demonstration for a test problem. We wanted to find a set of procedures and parameter values within our specified constraints which succeeded in learning a substantial amount of the structure available in the input. Given this reference point, we could then use the system to explore the effects of varying the learning parameters. This report summarizes our success in constructing a system which meets this initial design goal. Our primary success criteria is the number of target rules the system has learned over a given time period. In addition, we are interested in attentional learning, that is, discovery of the predictive features.

Descriptive Analysis

The first method of evaluating rule learning is examination of the set of rules discovered after a sig-

nificant learning exposure. We can ask how many universal, contrastive, and exception-based rules were discovered and what the number and nature of other, non-target rules were. Below, we describe results from one simulation run. It had 40% exceptional objects in the object base and a minor cycle of 400 iterations. We report the strong rules found by the system during learning, from among the 20 rules stored.

After **20 cycles**, the system had already learned rules about birds, fish and whales. It had also learned that all things breathe. Of the nine rules then, one is universal, five are contrastive, and three are exceptional. There are no overly specific rules in the top set.

(Feathers) ⇒ Wings
(Water) ⇒ Fins
(Scales Water) ⇒ Fins
Fins) ⇒ Water
(Fur Fins Water 1000) ⇒ White
(White Fur Water 1000) ⇒ Fins
(Feathers Wings) ⇒ Air
(White Fur Fins Water) ⇒ 1000
Nil ⇒ Breathes

After **40 major cycles**, the system learned about penguins. While the system had previously learned that feathers and wings imply air, it now knows that if the animal is dressed in what appears to be a tuxedo, we can expect to find it on land. Also notice that there is now an extraneous piece of information — (blue color) — in the second new rule. The frequency of blue fish hasn't changed, but the salience of color has increased, giving strength to overly specific rules such as this one.

(Tuxedo Feathers Wings) ⇒ Land
(Scales Water Blue) ⇒ Fins

By **50 cycles**, the system had learned about relations among all the contrastive features, had predictive rules for each type of exception, and knew the universal property of breathing. Though the system had not learned all possible predicitively valid rules, it had learned universal, contrastive, and exception-based rules.

(Legs Land) ⇒ Fur
(Brown Wings Land) ⇒ Fur

Two new rules were added at 100 cycles.

(Wings Air) ⇒ Feathers
(Fins) ⇒ Water

Good mastery of exceptions and contrasting classes was widespread across runs with different levels of exceptions and various parameters. Within the present system configuration, universal rules were only learned with 40% exceptions in input. Exceptions are important for successful generalization since our conservative generalization mechanism is driven by dropping conflicting features, and conflicts are only found in exceptions. This conservatism also means that incorrect generalizations are never found.

Performance Statistics

During the course of each learning experiment we collected data on how many of the contrastive and universal rules the system has learned. We do not include exception-based rules here because the set of possible exception-based rules is so large. There are a total of 27 contrastive and 2 universal rules. The 27 contrastive rules include all combinations of predictive relations among body covering, habitat, and limb type. These form an interlocking and redundant set of predictions, for example, predictions that wings and feathers imply flying; wings imply flying; and feathers implies flying. Thus, percent of rules learned provides a quite conservative measure; with respect to the training domain, predictive success could be perfect with 9 of the 27 rules.

Figure 1 shows the results of this data collection from runs with 0%, 20%, 30%, and 40% exceptions. Each line here is the average of two runs with identical parameters but different random factors in selection of objects to observe and rules to test. Given that there are exceptions at all, learning is slowed with a higher proportion of exceptions. However, a different pattern holds in the runs with no exceptions. In these runs, the system can only learn a third of the rules, and performance quickly moves to this level. Our generalization mechanism is very conservative. Generalizations are only introduced when a more specific rule is wrong. When there are no exceptions, rules about a pair of features, such as wings and feathers implying air, are never generalized further; wings implying air is never produced. Thus, without exceptions, the system quickly learns each of these nine predictive contrasting rules. This is sufficient to correctly predict all the contrastive regularities

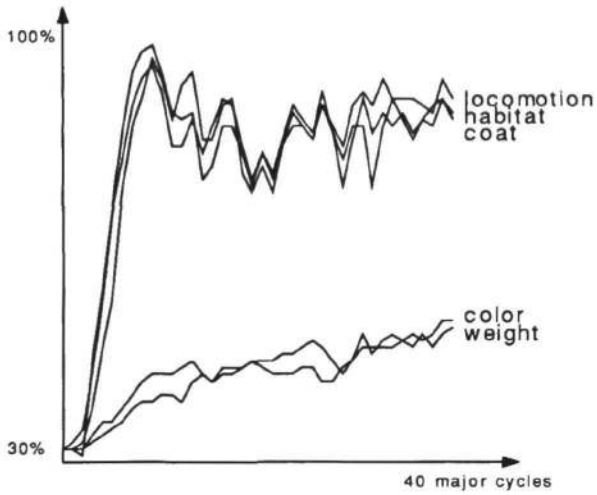


Figure 2: 20% Exceptions

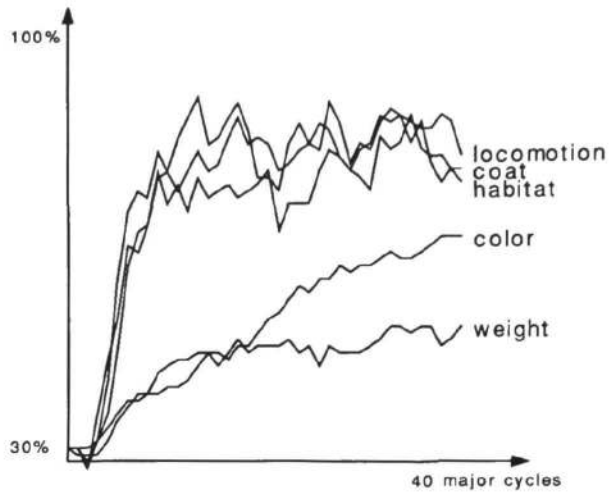


Figure 3: 30% Exceptions

found in the learning set and is another indicator of the conservative nature of our scoring rule as percent of all contrastive rules.

Attentional learning

In addition to learning about predictive rules, the system also learns what features are predictive. This information is an important part of making knowledge about the structure of the observed objects explicit. It is important information to use in explaining transfer (negative or positive) in learning new problems and may also facilitate learning within one problem. Figure 2 and Figure 3 show the salience of features in the 20% and 30% exception conditions whose rule learning curves appear in Figure 1. The constant (low) salience of the two universal features is not shown. Attentional learning is fast and produces sharp separation of the contrastive and idiosyncratic features with 0% (not shown) and with 20% exceptions. The system quickly learns which features are the best predictors. As the exception level increases, the contrastive rules are less and less reliable and the exceptional rules become the best predictors. Since color is a distinctive feature for each exception and is required (in combination with other features) for predictive success, color salience rises over the course of learning. With 40% exceptions (not shown) color ranks as the best predictor after about 40 learning cycles.

Summary

We summarize first the most important weaknesses and then the strengths of the current project. Our system requires that all objects be representable in the same set of features. However, in many learning problems the features relevant to objects in one subdomain are not relevant in another. Hence, the system cannot learn domains where features, not just feature values, vary. More fundamentally, there are limits to a representation based solely on conjunctions of observable feature values; while sets of conditional rules using such feature vectors are a fairly powerful form of representation, we do not believe it is sufficient to account for induction and inference using category knowledge. We need to add explicit representations of categories, not just the corresponding predictive rules. Our current work includes some procedures for gathering the information collected in sets of contrastive rules and building a semantic network of explicitly represented categories. These procedures for adding categories are very ad hoc and will require much more rigorous development. The most serious flaw in the present system in achieving its initial goals is our generalization mechanism. The system relies heavily on exceptions for learning general rules — this results in overly conservative learning. Stronger generalization procedures will be particularly important in learning a system of hierarchical categories with more than one level of contrastive categories.

One major strength of the current project is the problem analysis: systems of predictive rules and categories can be learned from observation of ex-

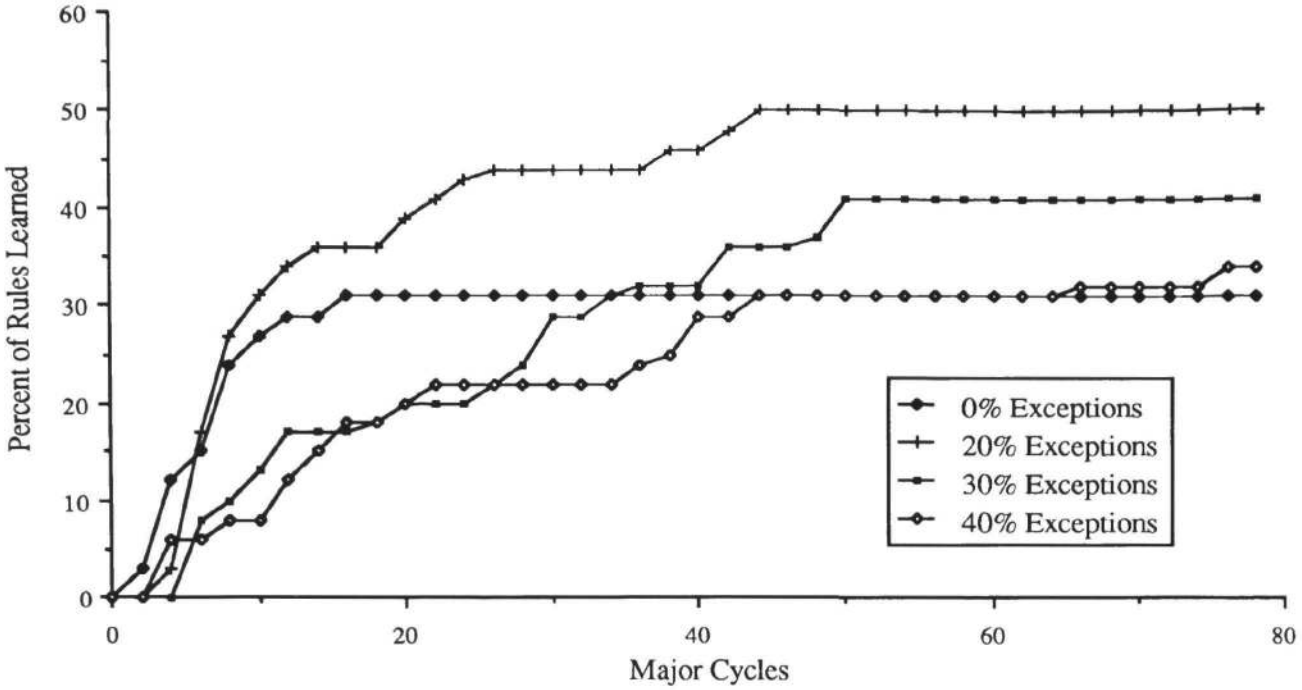


Figure 1: Rule Learning Curves with Focused Sampling

amples by comparing predicted and expected feature values. We identified three types of predictive rules and three possible ways that features can covary. We use our classification of features into universal, contrastive, and idiosyncratic categories to guide the design of our learning procedure. Specifically, we use feature salience and variability as well as predictive success and specificity to guide learning. Second, in applying this task analysis in the current simulations, we have a learning system which meets the design criteria specified initially and which successfully learns contrastive, exception-based, and universal rules. Finally, our simulation provides a flexible tool for further research. It allows us to modify components of the learning system and test their effects on different aspects of learning.

References

[AKB79] Anderson, J.R., Kline, P.J., & Beasely, C.M. (1979). *A general learning theory and its application to schema abstraction.* In G.H. Bower (Ed.), The Psychology of Learning and Motivation, Vol.13. New York: Academic Press.

[Billm87] Billman, D.O., Richards, J., & Heit, E. (1987). Abstraction of correlational rules in implicit concept learning tasks. in review.

[Billm87b] Billman, D.O., Heit, E., & Dorfman, J. (1987). *Facilitation from clustered features: Using correlations in observational learning.* In The Ninth Annual Conference of the Cognitive Science Society. Hillsdale, NJ: Erlbaum.

[Billm87c] Billman, D.O. & Heit, E. (1987). Observational Learning From Internal Feedback: A simulation of an adaptive learning method. Cognitive Science. In press.

[Garne74] Garner, W.R. (1974). *The Processing of information and structure.* Hillsdale, NJ: Erlbaum.

[Holla75] Holland, J.H. (1975) *Adaptation in Natural and Artificial Systems.* Ann Arbor: The University of Michigan Press.

[Holla86] Holland, J.H. (1986). *Escaping brittleness: The possibilities of general-purpose learning algorithms applied to parallel rule-based systems.* In R.S. Michalski, J.G. Carbonell, & T.M.Mitchell (Ed.), Machine Learning. Palo Alto: Tioga Press.

[Rosch78] Rosch, E.H. (1978). *Principles of categorization.* In E.H. Rosch & B.B.Lloyd (Eds.), Cognition and Categorization Hillsdale, N.J.: Erlbaum Publishers.

[Rumel86] Rumelhart, D. & McClelland, J. (1986). Parallel distributed Processing. Cambridge, MA: Bradford Books/MIT Press.

ON THE APPLICATION OF MEDICAL BASIC-SCIENCE KNOWLEDGE IN CLINICAL REASONING: IMPLICATIONS FOR STRUCTURAL KNOWLEDGE DIFFERENCES BETWEEN EXPERTS AND NOVICES [1]

Henny P.A. Boshuizen, H.G. Schmidt & Lorence D. Coughlin°,

University of Limburg, Maastricht, The Netherlands and
°McGill University, Montreal, Canada

Knowledge structures in memory and knowledge application are concepts which are closely related in theories on expertise in semantically rich domains. Glaser (1986) takes an extreme position by hypothesizing that novices' failure to solve a problem can be attributed solely to structural aspects of their knowledge which prohibit activation. This claim is supported by research findings of Allwood & Montgomery (1981), who showed that novices, while solving problems, have difficulties in activating relevant knowledge, and even seem unaware of their possessing that knowledge. In order to become applicable in problem-solving situations, knowledge must be conditionalized.

Lesgold (1984), applying Anderson's (1982) theory on learning of cognitive skills to medicine proposed that medical basic-science knowledge, initially declarative in nature, becomes tied to the conditions under which it can be applied, is compiled and tuned to medical practice.
Feltovich and Barrows' (1984) model of medical expert knowledge can be perceived as the final stage in this developmental process. They propose a multi-layer model in which medical basic-science knowledge becomes integrated into what they call "illness scripts". In their view, a physician who is confronted with a patient attempts to generate a description of the patient's major malfunctions (the Fault), the signs and symptoms which (s)he suffers from (the Consequences of the Fault) and how the patient's condition came to be (the Conditions that Enabled the Fault). For instance, sex, age and weight are conditions in the patient, associated with several kinds of diseases. Increasing age is associated with a higher incidence of cancer, whereas cardiovascular diseases are more commonly found in males and in people suffering from obesity. Being overweight, over 45 years of age and a female are associated with bile stones. Enabling conditions as defined by Feltovich & Barrows (1984), impose constraints on the set of plausible hypotheses about the Fault causing the patient's signs and symptoms. That is why bile stones will more easily be expected in a 55 year old lady who is 15 kg overweight, than in a thin young man, although both may present suffering from an attack of abdominal pain. The knowledge applied in identifying Enabling Conditions, Faults and Consequences, may be part of several medical subdomains and specialties like epidemiology, microbiology and internal medicine. Feltovich and Barrows claim that in such an illness script, basic-science knowledge plays a important and integrating role. Its causal nature determines how variations in patient findings can be accounted for.

Evidence for this proposition is provided by Lesgold, Feltovich, Glaser & Wang (1981) in the domain of radiology. They found that knowledge becomes more and more accurate in the course of development toward expertise, and that knowledge applied by experts in characterizing chest X-rays was far more extensive, detailed, relevant and goal-oriented as compared with intermediates' and novices' knowledge. Other authors, however, notably Boshuizen, Schmidt & Coughlin

[1]Preparation of this paper was made possible by a grant (#6266) of the Dutch Foundation for Educational Research (SVO) to H.G. Schmidt and H.P.A. Boshuizen.

(1987) and Patel, Evans & Groen (1988), suggest that basic-science knowledge plays a less dominating role in expert performance in medicine, while it does play a role in the earlier, intermediate stages. Their suggestion is based on the relative absence of references to basic science concepts in explanations provided by experts when asked to describe the pathophysiological processes underlying signs and symptoms displayed in a clinical case. Thus, two conflicting hypotheses about the role of basic-science knowledge in expert performance can be identified in the literature.

In this paper, research on the role of basic-science knowledge in clinical reasoning will be reported. The question of interest is whether or not basic-science knowledge plays an increasing role in the development of expertise. In addition an attempt will be made to determine at which stage in the development towards expertise illness scripts emerge as organizers of patient information. In order to investigate these issues two experiments were carried out. In Experiment 1 an explorative, qualitative approach was taken, utilizing four subjects with different degrees of expertise. These subjects were asked to think aloud while processing sequentially presented patient information. In Experiment 2, results of Experiment 1 were verified in a sample of 24 subjects, using both qualitative and quantitative procedures.

EXPERIMENT 1

Experimental Method

Subjects

Four subjects at different levels of medical expertise participated in this study: one novice (a second year medical student); two intermediates ("I-1", a fourth year student nearly finished preclinical training; and "I-2", a fifth year student who had finished both primary care and internal medicine clerkships) and one expert (a family physician with four years of experience).

Procedure

The subjects were presented with a case of chronic relapsing alcohol-induced pancreatitis with minor pancreatic insufficiency. The case describes a 38 year old, unemployed male with a history of neurotic depressions and alcohol abuse. One year earlier, he had an attack of pancreatitis, and now calls the family physician with a complaint of severe, boring pain in the upper part of the abdomen.

The case was presented on 48 typed cards, each containing one or more items of information characterizing the patient, the present history, physical findings or lab findings.

The subjects were asked to think aloud while being presented with the cards in a sequential fashion and to provide a differential diagnosis at the end. Sessions were recorded on tape and verbatim transcripts were produced.

Analyses

The Feltovich & Barrows model was used as an analytic tool. First, the original case, as presented to the subjects, was analyzed using the Feltovich and Barrows categories. For each case item, it was decided whether or not it fit with the pancreatitis script. Next, each script item was assigned to one of the model's categories. Subsequently, the four think-aloud protocols were analyzed counting the number of links in reasoning chains in which case items were clustered together, and the number of rules connecting concepts in a chain. A link in a chain is defined here as the association between two case items constructed by the subject. These links were classified utilizing the Feltovich and Barrows model.

For instance, when the expert is confronted with the first Consequence item (#8):

Complaint : "continuous pain in the upper part of the abdomen, radiating to the back"

He responds:

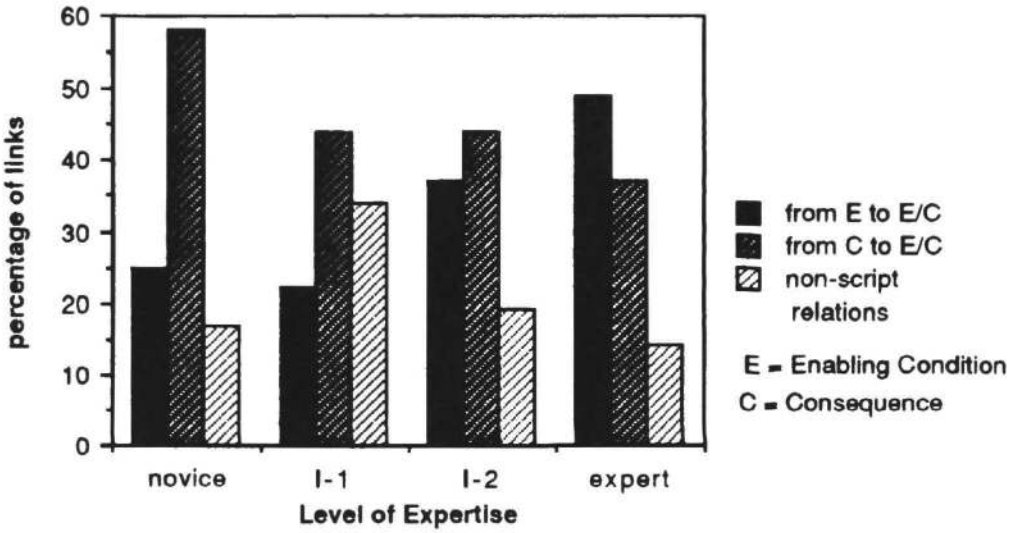


figure 1. Percentage of links between case items

"Well, when I visit a person (7) with an acute, continuous pain in his abdomen, radiating to the back (8), who suffered from pancreatitis a year before (5), where I don't know for sure whether or not he is still drinking after the Refusal cure (6), but I do know that his mental problem still exists (2), well I think that I'll think first: how is his pancreas? how is his liver? and that -because of his age (1)- other malignant processes in the abdomen are less probable." (relevant item# are between brackets.)

Here, the expert is linking an Enabling Condition (#7), Consequence (#8) and four other Enabling Conditions (#1,2,5&6) into a chain. Thus five links emerge from this excerpt: E #7 - C #8, C #8 - E #1, E #1 - E #2, E #2 - E #5 and E #5 - E #6. These links are classified as E-to-C, C-to-E and 3 E-to-E links.

In order to investigate the type of knowledge subjects were using in chaining the case items, rules were abstracted from the protocols. Rules were indicated in the protocols by expressions like: "<...> indicates <...>", "<...> makes me think of <...>", or "if you think of <...>, you always have to think of <...> as well". The abstracted rules were represented as IF-THEN statements. In the example above, that was derived from the expert's protocol, two complex rules can be identified:

IF	(house call & acute, continuous pain in upper abdomen & pancreatitis yr before & may drink alcohol & mental problems)	THEN	(pancreas or liver)
IF	(house call & acute, continuous pain in upper abdomen & pancreatitis yr before & may drink alcohol & mental problems & his age (38))	THEN	(less probable) (malignancy in abdomen)

Results and Discussion

Figure 1 summarizes the results of the analysis of the nature of the chains between case items found in each of the four protocols (E-to-E and C-to-E chains were take together as were C-to E and C-to-C chains.) The data show that non-script chains decrease from the intermediate to the expert levels, suggesting an increasing skill in deciding which patient findings are important in the final diagnosis. Furthermore, they show that reasoning chains, starting from a Consequence (a sign or a symptom) decrease with increasing expertise, whereas reasoning from Enabling Conditions to other Enabling Conditions or to Consequences (denoted E/C in figure 1) increases with experience. These data suggest an increasing use of this kind of knowledge in clinical reasoning.

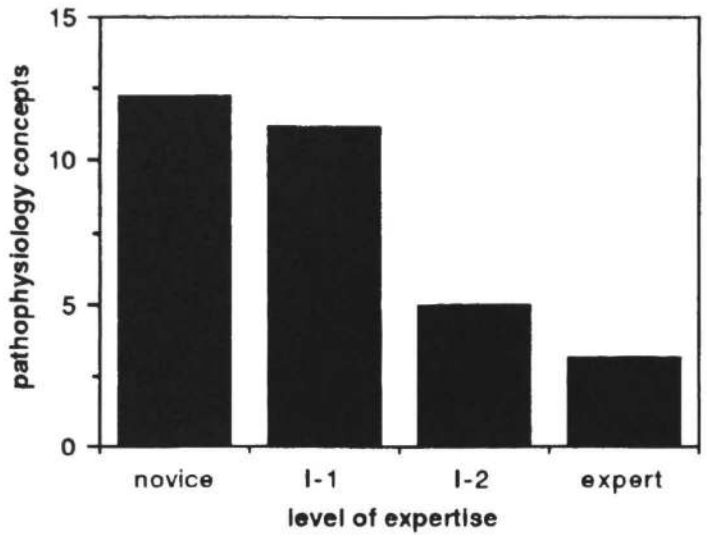


figure 2. Percentage of basic science concepts identified in the rules

In all subjects, in most rules that are abstracted from the protocols, case items are linked to a hypothesis about the patient's Fault. All subjects used Consequences to generate an underlying Fault, but the expert and subject I-2 in addition, displayed rules concerning relations between Enabling Conditions and a possible Fault. Again, indicating an increasing use of this kind of knowledge.

In summary, these results suggest that all subjects apply illness-script knowledge to interpret the nature of a patient's disease as is suggested by Feltovich and Barrows' model of medical expertise. With the development of expertise a shift can be observed from reasoning from Consequences to reasoning from Enabling Conditions in order to make sense out of a problem. It is particularly interesting to note that novices already apply illness-script knowledge before all relevant basic-science knowledge is acquired.

Figure 2 contains percentages of basic science concepts extracted from the rules applied by the four subjects. The data suggest that the application of basic-science knowledge is primarily characteristic of lower levels of expertise (the novice and subject I-1). In the expert protocol, hardly any references to basic science are made, indicating that basic-science knowledge does not play an overt role in expert clinical reasoning. A straightforward interpretation of this finding might be that the application of basic-science knowledge is characteristic of novice problem solving. The data suggest that the nature of the knowledge used by the expert in diagnosing a case is essentially illness-script related knowledge, whereas his basic-science knowledge has become inert (Bransford, Sherwood, Vye & Rieser, 1986) or even rudimentary.

However, an alternative interpretation is possible. Lesgold's (1984) theory suggests expert basic-science knowledge to be compiled and integrated. And since Ericsson & Simon (1984) assume that the application of compiled knowledge leaves no traces in think-aloud protocols, it may be possible that in experts basic-science knowledge plays its central and integrating role in a tacit way. This assumption however, would be at variance with data provided by both Lesgold, Feltovich, Glaser & Wang (1981) using the same thinking-aloud methodology, which suggests an explicit use of basic-science knowledge in clinical problem-solving.

In order to further investigate these alternative hypotheses on the role of basic-science knowledge in clinical reasoning, a second experiment was carried out. The knowledge actually applied in clinical reasoning was investigated utilizing the same on-line methodology as in Experiment 1. In addition however, the basic-science knowledge applicable to the case was elicited afterwards through a direct probe technique (Patel & Groen, 1986). It was expected that matching

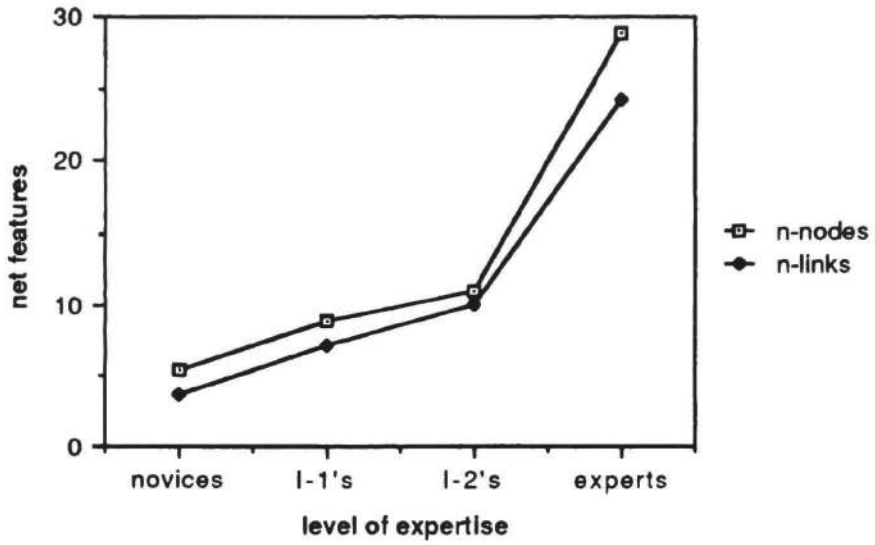


Figure 3. Average values of features of the semantic nets derived from the pathophysiological explanations of the case

information from these two sources would provide evidence as to whether expert basic-science knowledge is essentially rudimentary or inert, or whether it is compiled and integrated into illness-script knowledge.

EXPERIMENT 2

Experimental Method

Subjects

In Experiment 2, 20 subjects (6 novices, 4 fourth year and 5 fifth students, and 5 experts) participated. They were of the same levels of expertise as the subjects who participated in Experiment 1.

Procedure

Subjects were presented with the same case of pancreatitis using the same methodology as in Experiment 1. The subjects were asked to think aloud and to provide a differential diagnosis. Subsequently, they were asked to describe the pathophysiological process which, in their view, would explain the case.

Analysis

The think-aloud protocols were analyzed with respect to the type of knowledge that was applied in clinical reasoning. This was done in the same way as in Experiment 1. The pathophysiological protocols were analyzed using a methodology described by Patel & Groen (1986). Essentially, this procedure consists of a propositional analysis of the generated text. The resulting propositions are graphically presented in a semantic network, displaying the concepts and their relations appearing in the propositional analysis. The number of concepts and their connecting links were determined. Subsequently, the semantic networks were matched against the subjects' rule sets derived from the think-aloud protocols. The number of shared concepts was used as a measure for the overlap between semantic networks and rule sets.

Results and Discussion

Three alternative hypotheses emerged from Experiment 1.

1. Expert basic-science knowledge has become rudimentary. If this is the case, the experts'

pathophysiological protocols will be less elaborate than the intermediates', expressed in a smaller number of concepts.
2. Expert basic-science knowledge has become inert. If this is the case, no overlap will be found between expert basic-science and illness-script knowledge.
3. Expert basic-science knowledge has become integrated into illness-script knowledge. If this is the case, an increasing overlap between basic science and illness-script knowledge is expected.
These hypotheses are tested in a two-step approach. First, the extent of basic-science knowledge is investigated in order to test hypothesis 1. In figure 3, the results of this analysis are shown. Both analyzed net features: number of nodes ($F(3,16)= 3.811$, $p= .031$) and number of links ($F(3,16)= 3.567$, $p= .0379$) vary monotonously with increasing expertise. This indicates an ever increasing amount of basic-science knowledge about pancreatitis instead of a knowledge base that has become rudimentary.
As hypothesis 2 and 3 are mutually exclusive, they were jointly tested in the next step. It was found that the overlap between the semantic networks and rule sets increases monotonously from .149 in the novices to .562 in the experts ($F(3,15)= 19.854$, $p< .0001$). This finding demonstrates that in the more experienced and knowledgeable subjects basic-science knowledge is better integrated into the knowledge that is applied in clinical reasoning.
In summary, the results of Experiment 2 show that expert basic-science knowledge is more extensive than intermediate or novice knowledge. The increasing overlap between semantic networks and rule sets indicate that expert basic-science knowledge is neither inert nor rudimentary, but rather is integrated into illness-script knowledge. The results of Experiment 1, demonstrating the absence of this type of knowledge in clinical reasoning may be attributed to its compiled and integrated character.

GENERAL DISCUSSION

The results of Experiment 1 and 2, taken together, show two remarkable trends in the development of medical expert knowledge. The first experiment focussed on the application of illness-script knowledge, and the results suggest a fundamental restructuring of medical knowledge originating from various domains and subject matter. In clinical reasoning, this restructuring is evident in the increasing use of Enabling Conditions in the case description, and in a decreasing dependency on signs and symptoms. In addition, it was found that it was particularly the novice subject and not the expert, who overtly applied basic-science knowledge while reasoning about the case. Since Feltovich & Barrows (1984) assume that basic-science knowledge has an integrating role in expert clinical reasoning, this finding was unexpected. The second experiment showed that the absence of basic science concepts in expert clinical reasoning was not the result of a basic-science knowledge base which had become rudimentary. On the contrary, expert basic-science knowledge is more elaborate, but the concepts in the rules they actually apply in clinical reasoning coincide more often with pathophysiological knowledge suggesting that experts' basic-science knowledge can be run in compiled mode and is integrated into illness-script knowledge.
However, the finding that in the reasoning of all subjects, even the most novice, the application of illness-script knowledge is prominent, does not support Lesgold's theory (1984) of a compilation process starting from basic-science knowledge and resulting in illness-script knowledge. The results of the present experiments suggest that initially both knowledge layers are built up separately, and gradually become integrated during the developmental process toward expertise.

REFERENCES

Allwood, C.M. & Montgomery, H. (1981). Knowledge and technique in statistical problem solving. *European Journal of Science Education, 3*, 431-450.
Anderson, J.R. (1982). Acquisition of cognitive skill. *Psychological Review, 89*, 369-406.
Boshuizen, H.P.A., Schmidt, H.G. & Coughlin, L.D. (1987). *On-line representation of a clinical case and the development of expertise*. Paper presented at AERA-conference Washington, D.C.

Bransford, J.D, Sherwood, R., Vye, N. & Rieser, J. (1986). Teaching thinking and problem solving. *American Psychologist, 41*, 1078-1089.

Ericsson, K.A. & Simon, H.A. (1984). *Protocol analysis; Verbal reports as data*. Cambridge, Ma: MIT Press.

Feltovich, P.J. & Barrows, H.S. (1984). Issues of generality in medical problem solving. In: Schmidt, H.G. & De Volder, M.L. (Eds.), *Tutorials in problem-based learning; A new direction in teaching the health professions*, (pp. 128-142). Assen: Van Gorcum.

Glaser, R. (1986). On the nature of expertise. In: Klix, F. & Hofman, H. (Eds.), *Human memory and cognitive capabilities; mechanisms and performances, 2*, Amsterdam: Elsevier Science Publ.-North Holland.

Lesgold, A.M. (1984). Acquiring Expertise. In: Anderson, J.R. & Kosslyn, S.M. (Eds.), *Tutorials in learning and memory; essays in honor of Gordon Bower*. San Fransisco: Freeman & Comp.

Lesgold, A.M., Feltovich, P.J., Glaser, R. & Wang, Y. (1981). *The acquisition of perceptual diagnostic skill in radiology*. (Technical Report PDS-1). Pittsburg, Penn.

Patel, V. L., Evans, D. A. , & Groen, G. J. (1988). Biomedical knowledge and clinical reasoning. In: Evans, D.A. , & Patel, V. L. (Eds.), *Cognitive science in Medicine*. Cambridge, MA: MIT Press.

Patel, V.L. & Groen, G.J. (1986). Knowledge-based solution strategies in medical reasoning. *Cognitive Science, 10*, 91-110.

SIMILARITY-BASED AND EXPLANATION-BASED LEARNING OF EXPLANATORY AND NONEXPLANATORY INFORMATION

Woo-Kyoung Ahn and William F. Brewer

Department of Psychology & Center for the study of Reading
University of Illinois at Urbana-Champaign

ABSTRACT

We suggest that human learners employ both similarity-based learning (SBL) and explanation-based learning (EBL) procedures and that the successful use of these procedures is determined by the characteristics of the information to be learned. In a domain without underlying causal structure, multiple examples can lead to successful SBL, but not to successful EBL. In a domain with underlying causal structure, the use of appropriate background knowledge can lead to successful EBL, but not to SBL. A series of experiments was carried out in which a common initial passage was followed with a variety of different types of information (a second similar instance, a second contrasting instance, frequency data, or explanations). EBL occurred only when subjects had sufficient background knowledge and when the information to be learned could be causally structured. SBL occurred when there were multiple examples, even in domains without causal structure.

INTRODUCTION

Studies on concept formation and schema acquisition in Psychology have typically assumed that multiple examples are required for knowledge acquisition. These studies assume that knowledge acquisition is based on similarity-based learning (SBL), in which learners look for communalities and differences among examples and generalize the communalities and variabilize the differences (Anderson, Kline, & Beasley, 1979; Hayes-Roth & Hayes-Roth, 1977).

On the other hand, recent machine learning models of explanation-based learning (EBL), require only a single example for schema acquisition because the systems can analyze the causal structure of an example and generalize the explanation part of the example (Mooney & DeJong, 1985; Mitchell, Keller, & Kedar-Cabelli, 1986). Ahn, Mooney, DeJong, and Brewer (1987) presented psychological evidence that people can acquire a schema from a single example if they have sufficient domain knowledge or if the schema to be acquired has a causal structure. Ahn, Mooney, Brewer, and DeJong (submitted) also showed that EBL fails when people do not have sufficient domain knowledge or when the structure of a schema cannot be causally connected. These studies focus on EBL and do not show how this type of learning mechanism is related to SBL.

Recently, several researchers in the area of machine learning have attempted to integrate SBL and EBL (Danyluk, 1987; Kodratoff & Tecuci, 1987; Lebowitz, 1983, 1986; Pazzani, 1985; Pazzani, Dyer, & Flowers, 1986). These models make use of SBL or EBL in somewhat different ways. For example, Lebowitz's IPP looks for communalities of examples first and then tries to explain the communalities. Similarly, Pazzani's UNIMEM uses correlation first, since explanation is an expensive process. On the other hand, Pazzani et al.'s OCCAM uses prior causal theories in preference to correlational data.

In the present study, we outline a psychological theory of knowledge acquisition that uses both similarity-based learning and explanation-based learning procedures and carry out learning experiments designed to explore this hybrid approach.

EXAMPLE

The example used for this experiment was a description of a potlatch ceremony. This ceremony occurred among Indian groups of Northwestern North America; during the ceremony the host chief gave away valuables in order to improve his status. The following passage is a specific instance which includes the background knowledge necessary to understand the potlatch ceremony. The required background knowledge is given in brackets.

Guetela is a Kwakiutl chief and a descendent of Mamaleleqala. On July 13th, 1745, Chief Guetela invited four chiefs: Chief Namqic, a descendent of Dentalayo, Chief Qomoyue, a descendent of Mamaleleqala, Chief Laokoatx, a descendent of Wina, and Chief Tsamas, a descendent of Wina.

[The purpose of this ceremony was to increase Chief Guetela's status with respect to Chief Qomoyue who had the same ancestor. Both chiefs claimed the same family title because they both were descendents of Mamaleleqala. So Chief Qomoyue was entitled to compete with Chief Guetela for the status conferred by possessing the family title.]

[To be witness to the ceremony, chief Nemqic, Chief Laokoatx, and Chief Tsamas were also invited. These chiefs were invited because they were members of the same "moiety" as Chief Guetela's wife. Moieties are twofold divisions of a tribal group. Every individual is assigned to one of two moieties at birth, on the basis of the affilation of his or her mother. This means that Chief Guetela and his children were in opposite moieties.]

Before the ceremony, Chief Guetela and his tribe prepared for the ceremony by collecting as many blankets and canoes as they could afford. [Blankets and canoes were highly valued in this society. Chief Guetela wanted to give away these items because the more valuables Chief Guetela gave away during the ceremony, the higher his status became.] They also prepared smoked salmon and berries. Chief Guetela put on his best blue shirt with a raven on it. Dancers for the ceremony took raven and eagle masks from a copper box.

The guests arrived through the north gate of the village. Chief Namqic's shirt was orange, and Chief Laokoatx and Chief Tsamas's shirts were yellow. On these three people's shirts, an eagle was drawn. Chief Qomoyue was wearing a blue shirt with a raven printed on it. Chief Guetela's wife was wearing an

orange dress with an eagle printed on it and also a seashell necklace. [there are 4 additional paragraphs in the experimental passage]

Explanatory and Nonexplanatory Constraints and Variables

This passage was designed to contain four basic forms of information: constraints that are either explanatory or nonexplanatory and variables that are either explanatory or nonexplanatory. In a complex knowledge structure variables are slots which can be filled by different objects or agents. Constraints specify necessary properties of variables and necessary relationships between variables. Some constraints and variables have underlying explanations. For example, in the passage above, the information that the host chief's status improves after the ceremony is an example of an explanatory constraint because it provides an explanation for the purpose of the ceremony. The items given away by the host chief are examples of an explanatory variable because they play a role in the underlying causal structure, but the specific type of valuable object is not crucial. On the other hand, some constraints and variables are nonexplanatory. The information that the host chief looks at the guests before the dancing starts is a nonexplanatory constraint. There is no causal account of this action in our passage; it is simply a conventional behavior. The information that the guest chiefs arrive through the north gate is a nonexplanatory variable because it is not connected to the explanatory structure of the ceremony

ACQUISITION CONDITIONS

If an individual reads a single description of a specific instance of a potlatch ceremony that contains no explanatory information, our theory suggests that the learner will acquire little correct information from the passage since neither SBL or EBL processes will be possible. However, there are a variety of different forms of information that the learner can be given in a second passage, and our hybrid model of learning can be used to make a number of differential predictions in these cases.

SBL: Single Similar Instance

If a learner is given a second instance that is very similar to the original instance (i.e., has the same variables as the first one), then the learner should attempt to carry out SBL. With two instances of this type the learner will correctly assume that the repeated constraints are, in fact, constraints, but will incorrectly assume that the repeated variables are also constraints. For example, if the color of the host chief's clothing was actually a variable, but the same color occurred in both examples, then the learner will tend to think that this color is a requirement for a potlatch ceremony.

SBL: Single Contrasting Instance

If a learner is given a second instance that has each variable changed, then the learner should be able to carry out SBL for nonexplanatory constraints and variables. For example, if the color of the host chief's shirt was blue in the first example and it was red in the second example, the learner can infer that the color of shirts is a variable in a potlatch ceremony. For repeated constraints SBL will also succeed. For

example, if in both examples the host chief faces the guests before the dancing starts, the readers should use SBL to infer that this is a constraint in the ceremony.

SBL: Multiple Contrasting Instances

If a learner is given a series of instances that contain changed variables,then the learner should be able to use SBL to learn which aspects of the texts are variables and which are constraints. This condition is similar to the case of a single contrasting example, except that the larger number of cases should lead the learner to be more confident about the outcome.

EBL

If a piece of information is part of a causal structure and the learner is given the appropriate background knowledge, then the learner can use EBL to give an account of explanatory constraints and variables (Ahn, Mooney, DeJong, & Brewer, 1987).

METHOD

Procedure

Subjects read a first passage (i.e., the passage describing a single instance of a potlatch without any background knowledge). Then they answered all 16 yes/no questions, wrote justifications for their answers, and rated their confidence on a 5-point scale. After they finished answering all the questions, the subjects read one of the four follow-up passages and answered the yes/no questions a second time. Subjects were not allowed to change their answers for the first set of questions after they read their second passage.

Materials

The basic materials for this experiment consisted of an initial passage, four follow-up passages, and a series of yes/no questions designed to test what had been learned from the texts.

Yes/No Questions

We developed four types of yes/no questions There were questions based on the explanatory constraints and variables and on the nonexplanatory constraints and variables. One of the questions for explanatory constraints was, "After the ceremony will people think that the host chief has higher status than before?" One of the questions for explanatory variables was, "Will the wife of a host chief be happy if her husband gives away the family's drums?" One of the questions for nonexplanatory constraints was, "In this kind of ceremony, would it matter if the host chief did not look at the guests before the dancing?" One of the questions for nonexplanatory variables was, "In this kind of ceremony, is it necessary that the guest chiefs enter the village through the north gate?" There were four questions for each type, resulting in a total of 16 questions.

Initial Instance Passage

On the first trial, all the subjects received a passage which described a single instance of a potlatch ceremony without any added background knowledge. This passage corresponds to the example passage given earlier with the information in brackets omitted

Follow-up Passages

Similar instance passage. In this passage both the constraints and variables in the first initial passage were kept constant. For example, the guest chiefs left the village through the south gate in both passages.

Dissimilar instance passage. In this passage the constraints from the initial passage were repeated, but the values of the variables were changed. For example, the guest chiefs left the village through the north gate in this passage, while they left through the south gate in the initial passage.

Dissimilar instance with knowledge. In this passage the constraints from the initial passage were repeated, the values of the variables were changed, and the needed background knowledge was included.

Generic information with knowledge. This passage was written in generic form so that it gave roughly the frequency information that a learner would obtain after having read a very large number of individual passages. The passage began "The American Indians who lived in the Northwest part of the country frequently carried ot an interesting ceremony." For a nonexplanatory constraint it stated that "the host chief stood up and then dancers started entertaining the people," and for a nonexplanatory variable it stated that, "The shirts were all in various colors." This passage also included the relevant background knowledge.

Design

Fifty-two undergraduate students participated in this experiment in partial fulfillment of a course requirement for introductory psychology. There were four experimental conditions with 13 subjects in each condition. The subjects in the Similar Instance Condition received the initial instance passage and then the similar instance passage. The subjects in the Dissimilar Instance Condition received the initial instance passage and then the dissimilar instance passage. The subjects in the Dissimilar with Knowledge Condition received the initial instance passage and then the dissimilar with knowledge passage. The subjects in the Generic with Knowledge Condition received the initial instance passage and then the generic information with knowledge passage.

RESULTS

Table 1 shows the average percent correct for the four different types of questions in each condition. (EC refers to explanatory constraints, EV to explanatory variables, NC to nonexplanatory constraints, and NV to nonexplanatory variables). The numbers reported are the percent correct in each condition on the first trial and on the second trial.

TABLE 1. THE PERCENT CORRECT FOR EACH CONDITION ON EACH TRIAL

Conditions	Similar Instance		Dissimilar Instance		Dissimilar with Knowledge		Generic with Knowledge	
	1st	2nd	1st	2nd	1st	2nd	1st	2nd
EC	54	54	44	54	54	77	52	87
EV	44	46	48	48	54	81	50	77
NC	77	81	75	69	73	65	88	65
NV	29	27	29	81	33	58	13	39

The data provide considerable support for our dual-process theory of learning. We predicted that the subjects in the Similar Instance Condition would not be able to use SBL or EBL successfully and the data show little learning from the second example.

We predicted that the subjects in the Dissimilar Instance Condition would not be able to apply EBL, but would show SBL for the two nonexplanatory conditions. The subjects with dissimilar instance passages showed little learning in the explanatory conditions as predicted. The subjects showed considerable learning in the nonexplanatory variable condition, but they showed no learning in the nonexplanatory constraint condition This is not in keeping with our predictions. We are not sure what is going on in this condition, but perhaps the exposure to explanatory information temporarily changes their estimates of the likelhood that a nonexplanatory constraint is a true constriant. It is also possible that the subjects may be operating under a Gricean discourse rule (Grice, 1975) which leads them to believe that if an author chooses to mention something in the text it must be important and perhaps it is a constraint.

We predicted that the Dissimilar with Knowledge Condition would lead to successful SBL and EBL. All of these predictions are supported except that the nonexplanatory constraint information once again did not show the predicted learning.

We predicted that the Generic with Knowledge Condition would also lead to successful SBL and EBL. The results support the predictions except in the nonexplanatory constraint condition.

CONCLUSION

In this paper we propose that different learning mechanisms are successful depending on whether the learner has appropriate background knowledge or not and on whether the information to be learned is explanatory or not. Our data show that background knowledge is required to carry out EBL and to learn

explanatory information. Repetition is required to carry out SBL and to learn nonexplanatory items, but does not lead to the understanding and learning of explanatory information. This finding requires the development of theories which relate learning mechanisms to the character of the information to be learned.

REFERENCES

Ahn, W-K, Mooney, R. J., Brewer, W. F., & DeJong, G. F. (submitted) Schema acquisition from a single example. Unpublished manuscript, University of Illinois

Ahn, W-K, Mooney, R.J., DeJong, G.F., & Brewer, W.F. (1987). Schema acquisition from one example: Psychological evidence for explanation-based learing. Proceedings of the Ninth Annual Conference of the Cognitive Science Society, 50-57.

Anderson, J.R., Kline, P.J., & Beasley, C.M. (1979). A general learning theory and its application to schema abstraction. In G. H. Bower (Ed.), The Psychology of learning and motivation: Vol. 13. (pp. 227-318). New York: Academic Press.

Danyluk, A. P. (1987). The use of explanations for similarity-based learning. Proceedings of the Tenth International Joint Conference on Artificial Intelligence, 274-276.

Grice, H.P. (1975). Logic and conversation. In P. Cole, & J. L. Morgan (Eds.), Syntax and semantics: Vol. 3. Speech acts. (pp. 41-58). New York: Seminar Press.

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.

Kodratoff, Y., & Tecuci, G. (1987). Disciple-1: Interactive apprentice system in weak theory fields. Proceedings of the Tenth International Joint Conference on Artificial Intelligence, 271-273.

Lebowitz, M. (1983). Generalization from natural language text. Cognitive Science, 7, 1-40.

Lebowitz, M. (1986). Integrated learning: Controlling explanation. Cognitive Science, 10, 219-240.

Mitchell, T.M., Keller, R.M., & Kedar-Cabelli, S.T. (1986). Explanation-based generalization: A unifying view. Machine Learning, 1, 47-80.

Mooney, R.J., & DeJong, G.F., (1985). Learning schemata for natural language processing. Proceedings of the Ninth International Joint Conference on Artificial Intelligence, 681-687.

Pazzani, M.J. (1985). Explanation and generalization based memory. Proceedings of the Seventh Annual Conference of the Cognitive Science Society, 323-328.

Pazzani, M., Dyer, M., & Flowers, M. (1986). The role of prior causal theories in generalization. Proceedings of the Fifth National Conference on Artificial Intelligence, 545-550.

Causal Reasoning about Complex Physiological Mechanisms by Novices

Anoop S. Chawla
Vimla L. Patel

McGill University

Acquiring expertise in medicine involves the mastery of a wide repertoire of cognitive skills and and the accumulation and integration of a vast store of knowledge. It is generally agreed that in medicine, there are two distinct domains of knowledge; clinical knowledge and basic science knowledge. Clinical knowledge is primarily categorical and includes a classificatory scheme for disease entities and associated clinical findings. Basic science knowledge in medicine involves the organization of biomedical models at different levels of abstraction, (e.g., from the biochemical level to the organ systems level). Much of our research has addressed the issue of the of basic science in a clinical problem-solving context. We have found certain anomalies, namely that basic science knowledge does not necessarily support the diagnostic reasoning process (studies summarized in Patel, Evans & Groen, in press). Students use of basic science concepts frequently resulted in inconsistent clinical inferences that were actually counter-productive. In fact, the use of basic science in the causal explanations of expert physicians did not seem to improve the accuracy of diagnostic performance. This motivated us to examine in greater detail the process of how these basic science concepts are acquired and utilized. In this paper we investigate the application of complex concepts in the domain of pulmonary physiology.

Recently, a great deal of research has focused on the acquisition of scientific concepts (e.g., Pines & West, 1986). Many of the findings suggest that scientific concepts are inherently difficult to master because they typically require rather abstract formal representations that are not easily amenable for application in a particular domain (White & Frederiksen, 1987). The quantitative formal instantiation of these concepts is not generally consistent with students' naive intuitions about the phenomena that they purport to explain (diSessa, 1983). The evidence from many investigations clearly suggest that many students who have completed science courses acquire a knowledge of scientific concepts that cannot be used flexibly to interpret causal events and result in the acquisition of significant scientific misconceptions (Reif, 1987).

Feltovich et al. (1988) investigated medical students understanding of complex concepts in the domain of cardiac physiology. These studies document widespread misconceptions in the structure and function of a diseased heart. These misconceptions are developed early on in the learning process and become stabilized and entrenched in a student's diagnostic thinking. The authors argue that difficulty in attaining a deep level of understanding is often a result of reductionistic approaches to imparting the knowledge to the learner. Complexity is progressively introduced to early "scaffold" models, in order to facilitate an orderly acquisition of simplified models of complex biomedical phenomena. This oversimplification results in an in an inability to apply these concepts in "real-world" phenomena.

In this experiment, we continue on this trend towards investigating the nature of reasoning in biomedicine. We deviate slightly from our previous work in that we look at a problem which, although couched in a clinical setting, remains a problem of causal reasoning about

a mechanism and therefore fairly closed within the basic scientific sub-domain of physiology. As a result we propose that reasoning in the domain of physiology will have a certain correspondence to reasoning in *physical systems* domains such as physics (Mcloskey, 1983) and electronics (White and Frederiksen,1987), in the sense that there is a need for a qualitative causal understanding to enable the problem-solver to characterize a process and predict future states of a system. One important difference is that, like all biomedical sciences, pulmonary physiology is not a formal domain, in the sense that it is not possible to unambiguously state a set of principles that are sufficient to solve all clinically related problems (Larkin, 1981). The application of these concepts to clinical problems are therefore by abstraction and analogy (Patel, Evans & Groen, in press).

The Instructional Setting

The topic of pulmonary physiology is typically introduced to second-year medical students. The structural components that constitute the gas-exchange system are--air spaces- upper airway, bronchi, bronchiole, alveolar units, etc. The focus then shifts to the topic of pulmonary mechanics, in which the system is reduced to a *physical device* , and the behavior is explained in terms of the scientific laws governing fluid and gas mechanics-- *A difference in pressures drives a gas through a membrane.* This progresses to identifying the constraints of gas-exchange through a membrane, gas exchange through a lung unit, and exchange through the lung as a whole. The precise interrelationships are explicated through identifying the algebraic relations that hold. This is often without an alternate qualitative perspective. Class-lectures include an introduction to clinical cases; specifically, the measurable symptoms and the respective causal agents (the influence), and an identification of exactly what is happening at a a deep level by characterization in some physiological interpretation. Toward this end, many abstract concepts are introduced. For example, understanding the pathophysiological state of right heart failure may require an understanding of the relationship between pressure and flow and their relation to resistance in a compliant set of vessels.

METHOD

The subject pool consisted of 160 second year medical students at McGill University ,who were attending a series of lectures on cardio-pulmonary physiology. Following the lectures, the students were examined using a conventional multiple choice format. The lectures were monitored by our researchers and an additional essay question for the examination was devised in consultation with the instructor. The question was designed to assess the students' ability to comprehend selected concepts and use them in applied context. 40 of the students' essay questions were randomly sampled for our analysis. The stimulus text and question are presented below.

Table 1: Stimulus Text and Question Presented to Students.

A 30 year-old previously well non-smoker has a large pulmonary embolus which blocks the right pulmonary artery and half the vasculature of the left pulmonary artery. When he presents to the emergency department, he is very short of breath. His pO2 is 48 torr (N=100), and his pCO2 is 30 torr (N = 40 torr). A catheter was placed in his pulmonary artery and the pressure was 50/15 mmHg (N=25/8). The pressure in the right atrium was 12 mmHg (N < 8) and the cardiac output was 3.5 L/min. (N = 5-6 L/min.)
Discuss the factors which could have produced the hypoxia by predicting what would happen to dead space and ventilation-perfusion matching.

Cognitive Task Analysis

The task requires that subjects initially form a situational representation of the problem statement. The givens consist of a quasi-anatomical description of the patient's state and a set of values which represent deviations from the normal values (in brackets), reflecting the underlying pathophysiological perturbations. The first part of the task requires that subjects identify the process by which hypoxia (low concentration of Oxygen in the blood) results. They are primed to interpret the scenario through the concepts of dead space, which characterizes the state and ventilation-perfusion matching which, additionally, identifies a process of inadequate oxygenation of blood. The task is essentially, to build a causal chain from the given representation to the end state of hypoxia. These latter two concepts look at the relationship between ventilation and perfusion and therefore can be applied to various representations of the pulmonary unit for example the lung as a whole, the left and right lungs respectively, specific lung units, etc.

Data Analysis

A reference model was devised in consultation with three expert physicians and textual materials on the subject matter. The subjects' protocols were compared to a reference model by identifying; 1) the various concepts in the respective set of protocols; 2) the representation on which these are applied, by identifying locational cues in the text--" On the *left* side you have decreased perfusion, and normal ventilation..." which contrasts with a description such as-- "perfusion in the *entire lung* has decreased..."; and 3) concepts with a quantitative attribute, the specific value or direction of change-- " The V/Q ratio in the left lung has increased..."

The following represents a list of concepts that they may choose to bring to bear to the problem:

diffusion of gases through a membrane
shunt
oxyhemoglobin curve
O2-CO2 relationship
(inverse) relationship between CO2 and ventilation
compartmental model
dead space
relationship between ventilation and perfusion--V/Q
V/Q distribution
anatomical features of the respiratory system.

Table 2: Reference Model

There are four factors which produce hypoxia:

1)hypoventilation
2)shunt
3)ventilation-perfusion mismatching
4)diffusion problems

Hypoxia is present here as reflected in the low po2 value which is indicative of the concentration of oxygen in the blood as being low (not meeting demands of the tissues).

An embolus blocks all the flow to the right lung and half the left. This blood must then be accommodated by the 1/2 functioning left lung. That is, it gets forced over here. So in other words 1/4 of

the entire lung is receiving all the blood. The 3/4 of the lung which is now not receiving blood but is still being ventilated therefore represents physiologic dead space.

Normally the blood received in the still functioning lung is one quarter the cardiac output--1.5 L/min(6 L/min. divided 4 since it is one fourth the lung). In this case it is 3.5 L/min. so there is definitely increased perfusion. He is blowing off more CO2 as well which is reflected in his CO2 being below normal. So he is blowing it off faster than it is being delivered. And since it is only in the working region of the lung through which the gases can diffuse in or out (since this is the only place where air and blood meet), it seems that ventilation has increased as well,since ventilation is linearly related to co2. So v/q is normally 1-- same flow of blood as air in that region for which this v/q is applied; and in this case it is near that. So the v/q of this working lung is ok in the sense of measuring flow of air and blood through this region at a *gross* level.

Diffusion problems: 1)Blood may be going through at such a fast velocity now that by the time it reaches the end-capillary (there is a certain distance through which the blood and gas meet) the hemoglobin in the blood doesn't have time to pick up the oxygen diffusing through the membrane? That is, there is not full equilibration. This may be possible in cases where there is a massive amount of overperfusion. 2) Another secondary factor may be that the increased pulmonary pressure may cause edema leading to impaired diffusion. Shunt: If the increased volume of blood OPENED UP some blood paths which allow blood to flow from the venous side to the arterial side without it coming in contact with air (in alveoli) then this blood would not be oxygenated. This would be physiologic shunt. V/Q mismatch: There is a distribution of lung units with respect to the ventilation perfusion ratios i.e.,. Take any lung unit and see ventilation /perfusion. Most units in a normal lung have a ratio of one. In our case, there is a redistribution such that a lot more have a v/q ratio of < 1 with the increased perfusion and a lot less have a v/q of > 1 and of course there are less with v/q = 1 which is optimal for gas exchange. So the distribution is such that CO2 can still escape enough but not such that O2 can come in. CO2 and O2 behave differently from each other because hemoglobin hold oxygen and not CO2. There are different affinities for both of these by the blood as illustrated by the oxyhemoglobin curve.

Increased perfusion causes low pO2 i.e.,.--with normal perfusion, it happens to equilibrate at the point of 100 torr (mmHg). As you increase the perfusion you decrease the point of equilibration.

RESULTS and DISCUSSION

Out of 40 students, nineteen made the incorrect inference of 1/2 blockage in the left lung leading to decreased perfusion in the perfused portion of the left lung. Of related concepts, i.e., concepts apart from v/q mismatch and dead space, for which the students were primed for, 12 of 40 discussed the relationship between CO2 and ventilation, 5 of 40 discussed v/q distribution, 11 of 40 discussed the relationship between O2 and CO2 (often implicitly through mentioning the relationship between o2 and hemoglobin which makes it act differently than CO2).

With respect to other mechanisms being offered as an explanation to hypoxia, 17 of 40 attributed the hypoxia to edema resulting from increased pulmonary pressure; 2 of 40 mentioned the increased transit time (failure of complete equilibration-a diffusion problem), and 7 of 40 mentioned shunt as the significant factor.

Specific to the application of V/Q, some students chose to apply V/Q to the right and left lung respectively. That is, looking at the amount of ventilation in the right and the amount of perfusion. Since there is no mention in the stimulus text of a change in ventilation but the blockage infers zero perfusion, the two are combined to give a V/Q of infinity for this lung (some value / 0). Then looking at the left lung, some chose to apply it to the whole left lung while others partitioned this lung into the perfused and nonperfused portions. Others applied the concept at a global level to give a V/Q > 1-- reduced perfusion to ventilation.--

"The obstruction greatly decreases the blood perfusion, therefore resulting in mismatching of ventilation-perfusion ratio.." (S78)

The use of dead space was similar although many students limited the dead space to the right lung, failing to consider the nonperfused half of the left lung. Some students were global in their application by merely stating that overall dead space had increased.

In identifying the process of Ventilation-perfusion mismatch as the causal factor, it is essential that the reasoner determine that there are regions of overperfusion . We postulate that incorrect inference-- *partial blockage in the left lung causes decreased perfusion* arises for two distinct reasons:

1) There is a failure to consider that the blood normally going to the right lung must be accommodated by the unblocked left lung, illustrated by the following student protocols :

"The pressure in the pulmonary artery is increased since normal blood flow is impeded: blood is stuck there..."(S53).

"The embolus in the artery means that less blood is getting to the lung where it is needed...This [compensatory mechanism of hyperventilation] does nothing for the blood that is not reaching alveoli, since in this case most of the flow is in this position (i.e., not reaching air fluid spaces)..." (S82)

These excerpts illustrate that students often are incapable of maintaining the concept of circulation as a closed system, that is there is in some sense, no choice but for the entire blood flow to be routed to the unblocked pulmonary vessels. All the necessary information is not represented (carried over) when the focus of attention shifts to the left lung. In fact, this is largely present in protocols that partitioned the initial problem into a right/left scenario as opposed to a functional/non-functional distinction. The problem then, results from the *static interpretation* of concepts and the problematic use of *oversimplified representations* identified by Feltovich et. al. (1988)

2) If there are features in a sub-problem that have similar components to a previously solved sub-problem, then heuristics are used to infer relationships based on the respective change in values:

"The pulmonary embolus blocking the R.(right) pulmonary artery would cause an increase in physiological dead space and a ventilation/perfusion imbalance where essentially V/Q = infinity. This lung becomes non-functional in its capacity of eventual oxygenation of tissues. On the left side, there is also an increase in physiological dead space and an imbalanced V/Q ratio. There is less perfusion due to the embolus and so the V/Q ratio would increase. Assuming that you get half the perfusion expected initially, this would give a V/Q ratio of 2."(S97)

A *surface similarity mapping* is used in examination of different representations with overlapping sets of variables--In the right lung there is total blockage and zero perfusion, therefore since there is half blockage in the left lung, there is half perfusion.

In consideration of the variability in descriptions of causal relations that were identified, we postulate that explanation takes place by characterizing the problem state in terms of modifications to quantitative parameters in a piecemeal manner. These modifications are incorporated into discrete algebraic constraints with some apparent gaps between specific relationships in attempting to construct a coherent explanation. The solution is to use a heuristic to infer weak relations between these gaps, by observing the mutual deviations from the assumed "ideal" values among parameters in a system, in a *normally* functioning

state. These relations that are drawn support causal attribution, but fail to truly identify the pathophysiological process at the level of mechanism. This is a necessary condition for discriminating between different possible pathophysiological outcomes which is critical for practical implications such as therapy. Such modes of reasoning supercede more basic methods of envisioning a process, a form that better characterizes expert reasoning in such physical system domains (Kuipers, 1985; DeKleer and Brown, 1985).

In physiology, concepts such as ventilation-perfusion matching and dead space act primarily on *anatomically* based representations. For example, any attempt to determine dead space ventilation requires that one first identify the location of the lung being perfused and ventilated. This, we feel, is anatomical because it holds in the most efficient manner, the required information to reason about function and behaviour. Schematic representations serve a purpose in this respect (Larkin and Simon, 1987). There is, however, more than one such representation that must be invoked to understand a typical physiological phenomenon. How abstract concepts such as V/Q act on these represent the procedural component of the knowledge required to reason about a system. What is then needed is an explicit identification of how they relate to such representations apart from the models of dysfunction.

CONCLUSION

We have attempted to investigate reasoning about complex concepts in the domain of pulmonary physiology. With respect to instructional implications, our findings suggest that what is needed is an explicit identification of how a piece of knowledge contributes to comprehending the underlying pathophysiological state, as well as a specification of the cognitive nature of the representations that are used in reasoning about a problem. We argue that without a proper scheme for imparting this knowledge, reasoning proceeds in a non-optimal manner. Causal reasoning that employs models of *mechanism* is displaced by strategies that link knowledge components through more general heuristics in the process of constructing an explanation.

An effort must be made to identify the principles/attributes around which the concepts are most effectively organized in this domain. We speculate that most can be anchored in an *ideal* anatomical representation. Such a representation is more concrete and at the same time makes the relevant constraints explicit (for a large subset of problems) such that operations that require less cognitive effort, but concede to varying interpretations are avoided. They also serve as anchoring for other related concepts, which may rely on different anatomical representations but because it is the anatomical relationships that are linked, more abstract relationships are much more accessible and easily maintained.

References

DeKleer, J. & Brown, J.S. (1985). A qualitative physics based upon confluences. In D.G. Bobrow, (Ed.) *Qualitative rteasoning about physical sstems*. Cambridge, MA: MIT Press

diSessa, A.A. (1983). Phenomenology and the evolution of intuition. In Gentner, D. and Stevens, A.L. (Eds.) *Mental models*. Hillsdale, NJ: LEA.

Feltovich, P.J., Spiro, R., & Coulson, R.L. (in press). The nature of conceptual understanding in biomedicine: The deep structure of complex ideas and the development of misconceptions. In V.L. Patel & D.A. Evans (Eds.) *Cognitive science in medicine*. Cambridge, MA: MIT Press.

Kuipers, B.J. (1985). Qualitative Simulation. In D.G. Bobrow (Ed.) *Qualitative reasoning about physical systems*. Cambridge, MA: MIT Press

Larkin. J.H. (1981). Enriching formal knowledge; A model for learning to solve problems in physics. In J.R. Anderson, (Ed.), *Cognitive skills and their acquisition*. Hillsdale, NJ: LEA.

Larkin, J.H. & Simon, H.A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science, 11*, 65-99.

McLoskey, M. (1983). Naive theories of motion. In Gentner, D. and Stevens, A.L. (Eds.) *Mental models*. Hillsdale, NJ: LEA.

Patel, V.L., Evans, D.A. & Groen, G. (in press). Biomedical knowledge and clinical reasoning. In V.L. Patel & D.A. Evans (Eds.) *Cognitive Science in medicine*. Cambridge, MA: MIT Press.

Pines, A.L. & West, L.H.T. Conceptual understanding and science learning: An interpretation of research within a sources-of-knowledge framework. *Science Education 70*, 583-604.

Reif, F. (1987). Interpretation of scientific or mathematical concepts: Cognitive issues and instructional implications. *Cognitive Science*, *11*, 395-416.

White, B.Y. & Frederiksen, J.R. (1987). Qualitaive models and intelligent learning environments. In R.W. Lawler & M. Yazdani (Eds.) *Artificial Intelligence and education volume one*.. Norwood, NJ.: Ablex Publishing

A Comparison of Context Effects for Typicality and Category Membership Ratings

Leslie J. Caplan — National Institute of Mental Health

Robin A. Barr — National Institute on Aging

Introduction

In the last few years, the demonstration of context effects on people's judgments of category typicality (Barsalou, 1987; Roth & Shoben, 1983) has posed a major challenge to both classical feature and prototype theories of category representation. In these recent studies, the degree to which an item is considered to be a good example of a category depends on the context in which it is presented. This result has been taken to imply that category structure depends on situational factors and is, therefore, not the stable phenomenon which feature theories and prototype theories claim (Roth & Shoben, 1983). One possible conclusion which may be drawn from these apparent changes in category structure is that category representations are themselves unstable (Barsalou, 1987). Context effects have, however, provided support for exemplar-based accounts of category representation (Medin & Schaffer, 1978). According to these theories, category membership or goodness-of-example is determined by comparing items to category members in memory that are retrieved in response to the context. Because different contexts lead to the retrieval of different exemplars, an item that is encountered in different contexts will receive different goodness-of-example ratings.

However, some investigators have questioned whether goodness-of-example measures do indeed assess gradedness in category structure (Armstrong, Gleitman, & Gleitman, 1983; Barr & Caplan, 1985, 1987; Lakoff, 1987). According to these authors, gradients in typicality judgments do not necessarily reflect gradients in category structure or membership. They argue that it is easy to generate examples in which items are clearly members of categories, but are poor examples of the category. Thus, a penguin might be identified as clearly a member of the category Bird, but as a poor example of the category. If this is true, then context effects may only reflect differences in exemplar typicality, rather than differences in the degree to which an exemplar is viewed as a category member. For example, a change in context (e.g., from Montreal to the Antarctic) might change a penguin from a poor example of a bird to a

good example. In both cases, though, it remains clearly a bird. Therefore, context effects, as assessed by goodness-of-example ratings, may not reflect true changes in category structure.

In this experiment, we compared the context effects obtained for goodness-of-example ratings with those obtained for category membership ratings. If contexts truly change the structure of a category, then context effects should be equally apparent for both types of ratings. However, if they affect only an item's perceived typicality, but not its category membership, then context effects should be more apparent in goodness-of-example judgments than in category membership judgments.

We presented category members to subjects in two widely varying contexts. For instance, "ferry" and "raft" were each presented in the following sentence frames: 1) "The boys spent Saturday exploring the stream on the", and 2) "The cars had to reach the island by". The dependent measure we examined was the proportion of subjects who assigned higher ratings to exemplar A than to exemplar B in one context, and who reversed the direction of these ratings in the second context, such that exemplar B was assigned a higher rating than exemplar A. The existence of a reversal minimally implies a re-ordering of items along a scale - and, hence, a restructuring of the category.

Method

Subjects

Thirty-one adults served as subjects. All were between the ages of 18 and 45 years, had completed at least one quarter or semester of college, and were native English speakers. They were paid for their participation.

Apparatus and stimuli

Stimulus presentation and response recording were handled by a Macintosh microcomputer. Ten categories were used: Birds, Dogs, Flowers, Animals, Furniture, Boats, Vehicles, Weapons, Tools, and Toys. For each category, two exemplars were chosen. Three sentence contexts were presented with each of the exemplars in each category. One of the contexts, "They saw the", was used as a neutral context, but will not be discussed further. Each of the other two contexts was written to encourage the subjects to view one of the exemplars as a typical category member. Therefore, for each category, each exemplar was presented once with a congruent context, once with an incongruent context, and once with a neutral context. The order in which the 60 category-context-exemplar

combinations were presented was randomized differently for each of the two types of judgment.

On each trial, the following information was presented on the computer screen. The seven-point rating scale was presented at the top. In the category membership condition, the "7" was labeled "clearly a member" and the "1" was labeled "clearly not a member", after Barr & Caplan (1987). In the goodness-of-example condition, the "7" was labeled "very good example", and the "1" was labeled "very poor example (or not a category member)", adapted from Rosch (1975). Below the scale, subjects saw a context sentence ending in a category word. Below the sentence, subjects saw instructions to consider how well a particular exemplar matched their idea or image of what the category word had referred to in the previous sentence. At the bottom of the screen, a sentence requested them to rate the category word. In the category membership condition, they were asked to rate how clearly the item was a member of the category. In the goodness-of-example condition, they were asked to rate how good an example the item was of the category.

Procedure

Subjects were tested individually. They were instructed that they would be making two different kinds of judgments about a set of category words. The first type of judgment was explained to them, and they were asked to rate members of two practice categories without any presentation of context. Then, they were shown a printed facsimile of what the screen would look like on each trial, and were asked to rate the exemplar given, keeping the sentence context in mind. Next, computer-presented trials began. The first three trials were practice trials. Following these trials, subjects completed the 60 experimental trials for the first type of judgment.

During the second half of the session, subjects were given instructions for the type of judgment which they had not made during the first half, and the distinction between the two types of judgment was pointed out to them. The procedure for the second half of the session was virtually identical to that of the first, with the exception that the nature of the rating scale was varied as described above.

The order in which the two types of judgment were made was counterbalanced across subjects. All of the categories, contexts, and exemplars used were identical for the two kinds of rating.

Results and Discussion

For each category, we calculated the proportion of subjects who demonstrated reversals in their ratings. A reversal was defined as a difference between the ratings of the two exemplars in one context which was in the opposite direction to the difference between them in the other context. When the difference equalled zero for either context, that subject's response was not counted as a reversal. The proportion of reversals was much higher for goodness-of-example ratings (mean = .77) than for category membership ratings (mean = .37), F (1, 9) = 110.39, $p <$ .0001.

Clearly, reversals of exemplar ratings with context were more likely to occur for ratings of goodness-of-example than for ratings of category membership. This finding suggests that previous investigators may have overestimated the magnitude of context effects, and that conclusions that category structure is unstable may not have been fully warranted.

Even in category membership ratings, however, reversals of exemplar ratings did occur. How might such reversals be explained? One possibility is that, even for category membership judgments, context effects reflect a re-structuring of the category with context (Roth & Shoben, 1983), and a corresponding change in category representation. Another possibility, however, is that although the extension of a category may change with context, its representation does not. We have recently (Barr & Caplan, 1985, 1987) presented a theory of category representation which can explain how a category's representation can remain unchanged while its extension changes. According to our account, a category is represented by features, each of which may be considered to be *intrinsic* or *extrinsic*. Intrinsic features are true of an object considered in isolation (for example, most individuals would represent the feature "has four legs" to be an intrinsic feature of dogs). Extrinsic features are relations which hold between an object and some other entity (for example, most individuals would represent the feature "chases cats" as an extrinsic feature of dogs). Members of categories which are primarily represented by intrinsic features are likely to remain category members across changes in context, because intrinsic features are tightly bound to entities. On the other hand, members of categories which are primarily represented by extrinsic features may change their membership with context. Under some contexts, the appropriate conditions may hold for the

extrinsic feature to be true of an object. In these cases, the object will be considered a category member. Under other contexts, however, the appropriate conditions will not hold. In these circumstances, the object will no longer be considered a category member. For example, suppose that "used to create" were the complete representation of the category Tool. A piece of driftwood lying on a beach, therefore, would not satisfy the appropriate relationship to be a tool. However, if it were used by someone to scrape a picture in the sand, it would become a tool. It is important to note that regardless of whether a category's representation relies more heavily on intrinsic or on extrinsic features, the underlying representation of the category (i.e., the properties which determine category membership) remains the same in all contexts,. However, the set of items which are considered to be category members should change more for extrinsically represented than for intrinsically represented categories.

The data we collected in this study provide some support for this model. The 10 categories that were used vary in the degree to which their meanings depend on relational (i.e., extrinsic) information, according to the ratings of a previous group of 15 subjects. Therefore, we were able to calculate the correlation between previous subjects' mean ratings of "relationalness" and the proportion of subjects who demonstrated reversals for this set of categories. Using the more stringent measure of category restructuring (category membership ratings), we obtained a significant correlation between the degree to which a category's meaning relies on relational information and the proportion of reversals obtained, $r = .67$, $p < .05$. A similar correlation for goodness-of-example ratings failed to reach significance at the $p < .05$ level.

These data suggest that context effects are more likely to occur for extrinsically represented than for intrinsically represented categories. One should, however, be cautious in interpreting these results. Because the contexts used varied among categories, it is possible that we had inadvertently selected the more powerful contexts for the more relational categories. Nevertheless, these results are consistent with our model's predictions.

In conclusion, this study demonstrated that context effects are less likely to occur for category membership judgments than for category typicality judgments, suggesting that previous estimates of the frequency or magnitude of context effects may have been exaggerated. In addition,

the residual context effects shown when the more stringent measure of category membership was used can be explained to a considerable extent by a theory that assumes that category representation remains constant across contexts. It remains to be demonstrated then, that context effects do indeed challenge the traditional view that category representation is invariant across contexts.

References

Armstrong, J. L., Gleitman, L. R., & Gleitman, H. (1983). What some concepts might not be. *Cognition*, *13*, 263-308.

Barr, R. A., & Caplan, L. J. (1985). Two kinds of feature? A test of two theories of typicality effects in natural language categories. In *Proceedings of theS eventh Annual Conference of the Cognitive Science Society* (pp. 181-189). Hillsdale, NJ: Erlbaum.

Barr, R. A., & Caplan, L. J. (1987). Category representations and their implications for category structure. *Memory & Cognition*, *15*, 397-418.

Barsalou, L. W. (1987). The instability of graded structure: implications for the nature of concepts. In U. Neisser (Ed.),*Concepts and conceptual development: Ecological and intellectual factors in categorization*. New York: Cambridge University Press.

Lakoff, G. (1987). *Women, fire, and dangerous things*. Chicago: University of Chicago Press.

Medin, D. L., & Schaffer, M. M. (1978). Context theory of classification learning. *Psychological Review*, *85*, 207-238.

Rosch, E. (1975). Cognitive representations of semantic categories. *Journal of Experimental Psychology: General*, *104*, 192-233.

Roth, E. M., & Shoben, E. J. (1983). The effect of context on the structure of categories. *Cognitive Psychology*,*15*, 346-378.

Naive Materialistic Belief: An Underlying Epistemological Commitment

Miriam Reiner, Michelene T. H. Chi, Lauren Resnick

LRDC, University of Pittsburgh

A considerable amount of research has been done in mechanics which identified students' alternative frameworks in physics (Clement 1982, Viennot 1979, McDermott 1984). A few studies have also focused on students' beliefs in electricity, energy, heat and light. These studies show, in general, that many "misconceptions" can be traced to students' prescientific beliefs about the world. Furthermore, such beliefs appear to be fragmented, incoherent, built in pieces (Di Sessa, 1984), and context bound (McClosky, Caramazza & Green, 1980; McDermott, 1984; Reif, 1987). Hence, this set of beliefs has a limited power of explanation, and of course, differs from the scientific theories held by the scientific community.

Although naive explanations for a given phenomenon appear to be very context sensitive, it seems possible that there is an underlying implicit set of epistemological commitments that are respected in the various specific explanations. Nersessian and Resnick (1988) have outlined such a set of underlying commitments for the domain of mechanics. These commitments include the presuppositions that motion requires an explanation, (because it represents a change in state), and that an adequate explanation of motion must be framed in terms of mechanisms in which agents act physically on objects (rather than in terms of relations among theoretical constructs, as in Newtonian physics). These presuppositions were shared by pre-Newtonian physicists, and the local explanations constructed by today's naive subjects seem to share key features with those of Aristotelian and impetus theorists.

In this paper we explore another basic commitment--*a naive materialistic belief*--held by students, as well as by some adults, that can explain many of the students' beliefs as reported in the research on alternative frameworks in physics. The following example illustrates what we mean by a materialistic belief. In every day expressions, when we say that something is an illusion we usually mean that it is not real. We consider a moving car seen in a hologram to be an illusion, or unreal, even though we see it as if it was real. The reason that we refer to the moving car in the hologram as "not real" is that it is not made of what we call in everyday life "real material". An examination of students' explanations concerning light, electrical circuits, heat, energy, and force, suggest that they conceive of these entities in terms of the properties and behavior of "real material". Thus, we propose that a basic materialistic conception may underlie students' beliefs and understandings about a number of physical concepts.

Materialistic Interpretation of Light

Our hypothesis of naive materialistic commitments arose while working with adult physics students and teachers on concepts of lights. We begin by reporting this work and then examine other research on

alternative frameworks in different domains for evidence of materialistic commitments.

Conceptual beliefs about light have been investigated by Anderson and Karqvist (1981), Jung (1981), and La-Rosa (1984). The main focus of that work was on sight, the propagation of light through space, and issues in optical geometrics, but little details were provided about what students think that light is. Anderson and Karqvist (1981) report that students believe that "something" (either rays of light emanating from the eyes, images, or "something which goes backwards and forwards between the eye and the book") enables vision. But these investigators did not deal with the question of *what a student means when he uses the word light, or ray, or other such terms.*

In our work (Reiner, 1987), we interviewed 12 senior physics students and physics teachers about light phenomena related to every-day situation. All of the subjects had learned the appropriate physics to answer the questions, but the findings showed that their interpretations about light phenomena reflected materialistic commitment. Each subject was asked 4 questions: two questions concerned common everyday phenomena (the disturbance on T.V resulting from a lightning, and rotation of the radiometer); the other two questions were those commonly asked in physics classes (interference and defraction patterns of light). Analyses of the students' explanations showed that the the notion of massive "particles of light" played a basic role in explaining both the every-day phenomena and the physics class types of questions. For example, the rotation of the radiometer was explained by most of them as a result of the force or momentum transferred from the massive particles of light to the black plates of the radiometer.

It is not clear whether the source of this kind of materialistic conception arose from physics instruction or it might be rooted in a naive, preinstructional view. In order to examine this, we interviewed a new sample of bright, 17-year-old Technion-bound students who have not had the theory of light before. 31 out of 32 students showed a materialistic belief. Of the various specific beliefs expressed, three were dominant:

1. 22% of the students believed that light was produced by a hot body, which radiates a kind of material--a fluid or a stream of tiny particles--that can fill the air. Colors of light, according to this belief, results from the combination of two fluids that have different colors.
2. 18% of the students believed that a hot body emits particles of different sizes. Each size has its own color. It is easy to change a particle's color from white to any other color, but very difficult to change another color to white-- the same way that a white fluid will get a darker color if mixed with a darker fluid. It is interesting to note that this is very similar to Newton's theory of light--a point that reminds us that materialistic views were dominant in physics until quite late in the field's development.
3. 22% of the students believed that a chemical reaction causes the emission of light particles, which then have the ability to fill space. Many of these students noted that particles could be slowed down by the air (friction?) , and that is the reason that after a certain distance the particles stop, which explains why the light (such as from a candle) only goes a certain distance.

These results suggest that our initial sample of 12 senior physics students and teachers did not gain the

materialistic belief from books or any other forms of formal instruction. Instead, they appear to have retained a naive materialistic way of thinking which is fundamental to pre-instructed but bright 17-year-olds. Thus, both the naive 17-year-old students and the senior physics students and teachers have "materialized" light in two possible ways:

1. Light itself is considered to be a type of material--either small balls, or a stream of flowing continuous substance;
2. Light is a property of an object, for example, an object shines, so that light does not exist if the hot object, or the reacting materials in a chemical reaction, do not exist.

Materialistic Conception of Electrical Circuits

Maichle (1981) found that more than 80% of the students in secondary school level conceive of electric current as a kind of substance with the properties of quantity, storability and consumability. Even at the university level, 40% of the students agree to such statements. As might be expected, most of these students confuse the terms of voltage and current (Cohen, Eylon, & Ganiel 1983; Maichle, 1981). They use the term electricity for both current and voltage. Sometimes voltage is used to mean the storage of current. In this case, both current and voltage are materialized. According to students, "electrical material" is used up by the bulb, or other electrical devices, and hence there should be less current "after" the device "uses" it. The "misconception" of storability and consumability is a result of looking at electrical current as if it was a material which is stored in the battery. The material flows as long as there is "plenty of it" in the battery, that is, until the battery is "empty". This view is confirmed by Reiner's (1987) finding that high school students studying electricity believe that electrons have to "arrive" at the bulb in order to make the bulb shine, and that the propagation of the electro-magnetic field is equated with the flow of electrons in the wire.

Osborn and Gilbert (1987) found similar conceptions both among 18-year-old students studying A-level physics (the highest physics level in Britain) and 7-13-year-olds who had received no formal teaching in physics. The fact that both the naive and the 18-years-old students, who have a relatively good background in physics, refer to the battery as a storage of material which is the cause of current, implies that the materialistic framework appears to be resistant to teaching.

To summarize, students appear to materialize electricity in two ways: they think of electricity as if it was a material which can flow from one place (such as the battery) to another (such as the bulb); and furthermore, this material has the properties of being consumable and storable.

Materialistic Interpretation of Heat

Erickson and Tiberghein (1985) claimed that students imply that heat is substantive in nature, and describe it as something that can be stored in objects, transferred from one object to another, and travel from one location in an object to another. They also claim that every day expressions like "Close the

window to keep the heat inside" (and the cold outside) or expressions heard in classrooms such as: "heat is gained or lost", "travels along a metal rod", indicate that it seems to be natural to describe heat as a type of materialistic substance which can cause predictable changes in other objects. A "caloric" conception of heat, in which heat is referred to as a fluid that has no mass has also been found among students by several researchers (Albert, 1978; Erickson, 1980; Tiberghien, 1980; Shayer & Wylam, 1981).

To summarize, from these findings, we suggest that heat has been materialized by students in two ways: Heat is viewed as a kind of material (e.g., a fluid) which can travel from one place to another; and furthermore, like electricity, heat can be stored in materials, and takes on the shape of the object in which the heat is stored in the same way that gas takes on the shape of the container in which it is stored.

Materialistic Interpretation of Energy

Duit (1981) reported that for the overwhelming majority of students, the term energy in every day language has the meaning of a general kind of fuel to be used by motors and machines. Energy can be stored and consumed. To gain further insight into the meaning of the word for students, they were asked to give examples of energy. The results show that 51% of the students refer to the concept of energy as "things". According to this research, energy is not seen by the majority of students as an independent entity, but rather as a property of material bodies.

We again suggest that energy is materialized in two ways: first, energy is a kind of fuel, and second, energy is a property of an object, and exists only if an object exists.

Materialistic Interpretation of Forces

If naive students have a commitment to materialistic explanations, they should have difficulties in explaining motion in the absence of an external contact force. This is indeed the case. For example, Clement (1982) reported that students believe that the motion of a body implies that a force was exerted on it. In the case of a coin tossed up in the air, they think that "the coin has a force". This force, according to Clement's students, is believed to be decreasing during the coin's motion upwards, until the gravitational force overtakes the "coin's force", and then the coin will reverse it's direction and fall down. Minstrell (1982) also reported that his students' notion of a moving body is one that "has a force in itself". Gunstone and Watts (1985) also reported that forces are explained as having to do with living things. Students talk for instance about "an object trying to fight it's way upwards against the will of gravity", which is very similar to Clement's results. In all three of these cases, force is viewed as a *property of an object*. The force does not exist unless the object exists. Force in this sense has no meaning independently of a materialistic object.

If force is viewed as a property of an object, then, force from a distance poses a special problem for

students. Although students use the term gravity, they do not understand that it can cause objects to fall. Instead, weight, and "falling down", are both perceived as two different properties of a materialistic body (Vicentini, 1983). "Falling down" is explained by the naive belief that a body falls because it 'wants' to be in it's natural place, which is "down". Vicentini-Missoni (1981) stated it as the "first law" formulated by students:"All heavy objects fall down, if nothing sustains them". This issue has been investigated by Nusbaum and Novak as well (19xx). They asked students to predict the motion of a body through a hole dug through the earth. Most of the students answered that the dropped stone will "fall down" through the center of the earth, and "fall down", out of the earth. The notion of gravity as a force directed to the center does not exist, and the motion of the stone is not related to the gravitational force towards the center, but rather is referred to as a property of the body. According to this belief, every materialistic object has the property of "falling down". The force from distance does not exist per se, unless it is related to a materialistic body. Thus both weight and the force driving objects to fall are transferred into *properties of a body.*

Another way of materializing force is by transfering the force into a contact force between the body and the air. Driver (1985) reported that some students believe that in the outer space there is no gravity because there is no air.Vicentino and Missoni (1981) reported that adults explain gravitation as : " The force of gravity is due to the air pressure". In this case the force from distance is transferred into a force of contact between two materialistic bodies. This is not a very new belief; Gassendi, a 17-th century scientist, conjecturing about the existence of a vacuum, remarked that no forces and no motion will be possible in the vacuum because of the absence of contact among bodies that could produce a force.

To summarize, we suggest that forces have been materialized in two ways: It is a property of an object, and a force is a contact between two (or more) materialistic bodies.

Conclusion

We have shown that a basic materialistic commitment underlies some of students' conception of key physics concepts. This materialistic commitment has 3 characteristics:

1. The entity itself is a kind of material (e.g., light is a massive particle or a kind of a fluid; electrical energy is equated with material fluid; and heat is caloric material);
2. The entity can be stored in material. There exist other objects whose role is to store the entity (e.g., light is stored in materials such as in the air which explains why reason we have light after sunset; electrical current is stored in the battery; and heat is stored in material objects; and force is captured in the body);
3. The entity exists only as a property of a materialistic object (e.g., light is a property of a hot body, or materials that react chemically; energy is a property of an object--the body has energy; gravitational force--falling down--is a property of a body; a moving body has force in itself, "Force of falling" is a result of the contact of two materials: air and the falling object);

The analyses reported here characterize the ways students conceptualize numerous physics concepts (light, electricity, heat, energy, and force) in terms of a basic underlying materialistic interpretation. The pervasiness and apparent robustness of such a fundamental conception across different physics concepts raise three important issues. First, if such a conception is so fundamental to the way we understand these concepts, how readily can instruction modify such a view. That is, what kind of instructional methods needs to be designed that can undermine such a basic epistomological commitment and produce conceptual change. Second, because such a materialistic view appears to be so inherent in students conception of the physical world, it seems to engender similar interpretations to related entities and concepts. For instance, students apply their conception of the properties of electrical current to the flow of energy in a wire. It is not clear how we can terminate the propagation of this kind of materialistic misconceptions. Finally, this research questions the validity of instructional approaches which use materialistic models as analogical tools. For example, do mental models of electricity such as flowing waters or teeming crowds distort our understanding of electricity by sustaining the materialistic view. Additional research is needed to address these issues.

REFERENCES

Albert, E. (1978). Development of the Concept of Heat in Children. *Science Education, 62*, 389-99.

Anderson, B., & Karrqvist, C. (1981). *Light and its properties.* (Ekna Report No. 8.). Translated into English by Gilian Thylander (1982).

Clement, J. (1982). Students' Preconceptions in Elementary Mechanics. *American Journal of Physics, 50*, 66-71.

Cohen, R., Eylon, B., & Ganiel, U. (1983). Potential difference an current in simple circuits: A study of students' concepts. *American Journal of Physics, 51*, 407-412.

Di Sessa, A. (1984). Learning about knowing. In E. Klein (Ed), *Children and Computers.* San Francisco: Jossey-Bass.

Driver, R. (1985). Changing perspectives on science lessons. In N. Bennett, & C. Desforges (Eds.), *Recent advances in classroom research. British Journal of Psychology Monograph.*

Duit, R., (1981). *Students' notions about the energy concept before and after physics instruction.* International Conference at Ludwigsburg, West Germany.

Erickson, G. (1980). Childrens' viewpoints of heat: A second look. *Science Education, 63*, 323-36.

Erickson, G., & Tiberghien, A. (1985). Heat and temperature. In R. Driver, E. Guesne, & A. Tiberghien, (Eds.), *Children's ideas in science.* Milton Keynes, England: Open University Press.

Gunstone, R., & Watts, M. (1985). Force and motion. In R. Driver, E. Guesne, & A. Tiberghien, (Eds.), *Children's ideas in science.* Milton Keynes, England: Open University Press.

Jung, W. (1981). Conceptual frameworks in elementary optics. In *Proceesings of the International Workshop on Problems Concerning Students' Representations of Physics and Chemistry Knowledge*, Ludigsburg, West Germany.

La Rosa, C., Mayer, M., Patriz, P., & Vincenti-Missoni, M. (1984). Commonsense knowledge in in optics: Preliminary results of an investigation into the properties of light. *European Journal Science Education, 6*, pp. 387.

Maichle, U. (1981). Representation of knowledge in basic electricity and it's use for problem solving. *In Proceedings of an International Workshop on: Problems Concerning Students' Representation of Physics and Chemistry Knowledge.* Padagogische Hochschule Ludwigsburg.

McCloskey, M. , Caramezza, A. & Green, B. (1980). Curvelinear motion in the absence of external forces: Naive beliefs about the motion of objects. *Science, 210*, 1139-1141.

McDermott. L. C. (1984). Research on conceptual understanding in mechanics. *Physics Today, 37*, 24-32.

Minstrell, J. (1982). Explaining the "at rest" condition of an object. *The Physics Teacher, 20*, 10-14.

Nersessian, N., & Resnick, L. B. (1988). *Epistemological obstacles to construction and inertial representation of motion.* Unpublished manuscript.

Nussbaum, J., & Novak, J. D. (1976). An assessment of children's concepts of the earth utilizing structured interviews. *Science Education, 60*, 535-550.

Osborn, R. J., & Gilbert, J. K., (1980). A method for investigating conceptual understanding in science. *European Journal of Science Education 2*, p. 311-321.

Tiberghien, A. (1979). Modes and conditions of learning. An example: The learning of some aspects of the concept of heat. *In Proceedings of an International Seminar on Cognitive Development Research in Science and Mathematics.* University of Leeds, Leeds.

Tiberghien, A. (1980). Quel rapport y a-t-il entre ce que les eleves "ont dans la tete" et ce qu'ils font ou disent? In Livre du Professeur 3eme coll Libres Parcours (Ed.), *Sciences Physiques*, (pp. 197-202). Hachette, Paris.

Reif, F. (1987). Interpretation of scientific or mathematical concepts: Cognitive issues and instructional implications. *Cognitive Science, 11*, p. 395-416.

Reiner, M. (1987). *Real time computer based analysis in physics laboratory, as a means for changing students' conceptual frameworks in physics.* An unpublished Doctoral Thesis.

Vicentini-Missoni M. (1981). Earth and gravity: Comparison between adult's and children's knowledge. *In Proceedings of an International Workshop on Problems Concerning Students' Representation of Physics and Chemistry Knowledge*. Padagogische Hochschule Ludwigsburg.

Viennot, L. (1979). Spontanous Reasoning in Elementary Mechanics. *European Journal of Science Education, 1*, 205-221.

Writing Expertise and Second Language Proficiency: Algorithms and Implementations?

Alister Cumming

Faculty of Education

McGill University

Anderson (1987) argues for a fundamental distinction between algorithmic and implementation levels of cognitive processing, as well as specific research methods to investigate them. The process-tracing study reported here (summarizing Cumming 1988) confirms the relevance of Anderson's distinction in a naturally occurring instance of human performance: more and less expert writers composing on different tasks in their second language. The results of the study are also consonant with theories distinguishing intelligent central processing from informationally-encapsulated knowledge of language (Fodor 1983, Chomsky 1988) or suggestions that literate expertise develops as a unique core intelligence (Gardner 1983, Bereiter & Scardamalia 1987).

Educated adults writing in their second language display a natural disjuncture between (1) the higher orders of planning, heuristic searches, and uses of complex representations which characterize expert thinking and (2) a greater or lesser facility for the rapid, non-deliberative processing of language, depending on their proficiency in the second language. They show differentiation between higher and lower orders of cognitive processing of the kind Anderson (1987) describes as algorithms and implementations. This differentiation is more visible with mature learners than in conventional studies of cognitive development which have focused on children, whose processes of linguistic, conceptual, maturational, and social development are necessarily interconnected.

Interestingly, though, greater writing expertise <u>or</u> second language proficiency appear to lead to enhanced writing performance in a second language -- suggesting there is a value for learning at (in Anderson's terms) either the algorithmic or implementation level. Knowledge at both levels seems necessary for optimal performance in this domain. However, writing expertise appears to be knowledge which is not tied to the domains of a first or second language. Characteristics of writing expertise appear to be enacted in similar ways across people's first and second languages, irrespective of their levels

of second language proficiency (Arndt, 1987, Edelsky, 1986, Gaskill, 1986, Jones & Tetroe, 1987).

APPROACH

Evidence of these distinctions was obtained by analyzing the performance of 23 young adults writing 3 composition tasks in their second language (an informal letter, an expository argument, and a summary of a popular science booklet). Participants were carefully selected to represent: 3 levels of writing expertise in their mother tongue (5 professional writers, 8 average students, and 10 basic writers); 2 levels of proficiency in their second language (11 at intermediate and 12 at advanced levels of English); a common mother tongue and cultural background (French-Canadians raised in Quebec); common levels of education (first and second years of university); and common motivations for learning their second language (all had moved to the same English-French bilingual university in Ontario to improve their English). These characteristics were verified through evaluation of participants' writing in their mother tongue, self-ratings of literate abilities, background questionnaires, and interview tests.

Three aspects of writing performance were assessed using a 3 (writing expertise) X 2 (second language proficiency) X 3 (tasks) factorial design. Qualities of the texts produced were rated for the effectiveness of their content, discourse organization, and language use. Decision statements were extracted from think-aloud protocols then analyzed in two ways. One, problem solving behaviors used to control writing processes were analyzed for the extent to which participants used heuristic search strategies. Two, the decision statements were analyzed to establish whether participants were attending to one, two or more aspects of their writing while making decisions. Aspects of writing were defined as gist, language use, discourse organization, intentions, or procedures for writing (following Scardamalia & Paris, 1985). Inter-rater and intra-rater reliability on these analyses ranged from .7 to .9.

DISTINCTIONS IN LEVELS OF COGNITIVE PROCESSING

Multivariate analyses revealed large main effects for the factors of writing expertise and second language proficiency on the ratings of text qualities and problem solving behaviors. Interestingly, there were no interactions between the factors. This suggests that writing expertise and second language

proficiency are psychologically distinct, supporting theories of different levels of cognitive processing or mental modularity. Participants had developed writing expertise and/or second language proficiency to different extents, each of which made separate contributions to their writing performance in the second language.

For the qualities of the compositions produced, a repeated measures MANOVA revealed main effects for writing expertise (F (2, 16) = 25.1, p <.0001) and second language proficiency (F (1, 16) = 53.8, p <.0001), without any interactions between the two factors. A similar analysis of decision making behaviors using heuristic searches (to resolve problems encountered while writing) also showed separate main effects for writing expertise (F (2, 16) = 29.0, p <.0001) and second language proficiency (F = 5.2, p <.04), without any interactions. Across tasks, performance on the more cognitively demanding argument and summary tasks consistently differed significantly from performance on the less cognitively demanding letter task. With increased writing expertise or second language proficiency, people tended to produce more effective compositions, make more extensive use of heuristic search strategies, and do this to a greater extent on more demanding tasks.

The complexity of mental representations people used for decision making was analyzed by contrasting the extent of references they made to either 1 or 2 (or more) aspects of their writing in protocol statements. A repeated measures MANOVA showed a main effect for writing expertise (F (2, 16) = 8.4, p < .003), non-significant effects for second language proficiency, and an interaction between the two main factors (F (2, 16) = 4.7, p < .03). More expert writers tended to refer to 2 or more aspects of their writing while making decisions, while inexpert writers tended to consider only 1 aspect. The interaction effect appeared to arise for people without high or low levels of writing expertise (i.e. the average students). Average students with lesser levels of second language proficiency acted like the inexpert writers, attending mainly to 1 aspect of their writing while making decisions. Average students with higher levels of second language proficiency acted like the expert writers, attending mainly to 2 aspects of their writing while making decisions.

Qualitative analyses of the think aloud protocols indicated that second language proficiency did not visibly affect the processes of composing in the second language. No consistent differences could be discerned between the reported thinking processes of people with greater or lesser second language proficiency. This confirms the widespread claim that people have little conscious access to knowledge of their second language (Krashen, 1982, Seliger, 1983, Carroll, 1985). It also lends credence to Anderson's (1987) argument about the limitations of research at the "implementation" level of cognitive processing.

Obvious differences were evident, however, in the approaches to composing displayed by the more and less expert writers. These differences appeared regardless of participants' second language proficiency. They correspond to findings in previous research on the cognitive processes of mother tongue writing (de Beaugrande, 1984, Bereiter & Scardamalia, 1987, Flower & Hayes, 1980). Expert writers displayed well-differentiated approaches to planning their discourse (as emergent or advance planning), using rhetorical scripts and goals to guide their writing, transforming their thinking flexibly to solve problems, and concerning themselves persistently with the qualities of their word choices.

In contrast, less expert writers displayed little control over their writing processes. They tended to plan in small, constrained units -- frequently asking themselves what to say next, because they had little sense of how to proceed with their overall discourse. Alternatively, they simply wrote down everything that came to mind, without assessing its value or quality. They displayed little concern for word choices.

OF WHAT VALUE IS THIS DISTINCTION?

Characteristics of expertise at the "algorithmic" level were reportable, consistently evident in think aloud protocols, and distinguished from processing at the linguistic level. This suggests they represent a level of cognitive knowledge which differs qualitatively from the lower levels of language "implementation". As such, they appear amenable to reflective awareness, self-regulation, learning, modelling, and instruction, as Anderson (1987) proposes, and others have demonstrated for the learning of higher-level literate behaviors (Bereiter & Scardamalia, 1987, Brown, Palincsar and Armbruster, 1984, Cumming, 1986). This knowledge might be called intelligent processing, in view of Gardner's (1983) definitions of literate knowledge as a specialized core intelligence.

In contrast, knowledge of a second language was largely beyond the awareness of individuals' attention during task performance. It is knowledge that participants had acquired over time and social interactions, correlating closely with their length of residence in the English speaking environment (.9 $p < .001$). As theories of second language acquisition suggest, people develop such knowledge through purposeful use of the language, progressively matching their behaviors and interpretations to standards salient in their social environment (McLaughlin 1987). But, as Fodor (1983) claims, such linguistic knowledge appears modularized and encapsulated -- and thus not amenable to much conscious manipulation. Nonetheless, expert writers obviously use their knowledge of a second language as they write, in much they same way as they do their first language.

For education, this distinction indicates that quite different learning curricula would be appropriate to foster the development of writing expertise and second language proficiency. For research into cognitive processing and knowledge, there is evident value in assessing adult performance on complex tasks in a second language. The natural disjuncture which adults display between their analyzable control over their expert thinking and their implicit second language proficiency make them suitable subjects to validate theories of modularity, levels of cognitive processing, and self-regulation.

WHERE DO THE LEVELS INTERACT?

Claims for distinctions in levels of cognitive processing beg the question of how such levels interact psychologically. A clue to how this might occur appeared in the persistent word searches and evaluations which characterized the thinking of more expert writers. This behavior -- also observed by Butler-Nalin (1984) in a comparison of second language and mother tongue students writing -- entailed simultaneous attention to language use, gist, and discourse structures; cross-linguistic comparisons; evaluations of alternatives; and finally resolution of a right choice:

> A model to... Ah, un modèle à analyser. A model to study.
> Not model. A very interesting... Not kinds. Very interesting...
> Not style. A very interesting... Ah, c'est pas un modèle.
> Cats are among others, a, un très intéressant, a very interesting... Ah, cats are a very interesting... Ah, case to
> study.

This pervasive behavior demonstrated the kinds of schematic searches, evaluations, retaggings and consolidations which Case (1985) claims are the main regulatory processes leading to intellectual development. Moreover, in doing this, expert writers were progressively verifying the truth correspondences between their thinking and language, a process which Johnson-Laird (1983) and Davidson (1984) claim are integral to human learning and intentionality. In order to generate alternative cross-linguistic word choices, it is necessary to make a priori equations across semantic, discoursal, and linguistic categories (Lipski 1978). In this way, expert writers appear able to integrate their knowledge of writing, their second language, and their mother tongue -- while at the same time learning from writing in their second language.

Davidson, D. (1984). Inquiries into truth and interpretation. Oxford: Oxford University Press.

Edelsky, C. (1986). Writing in a bilingual program. Norwood, N.J.: Ablex.

Fodor, J. A. (1983). The modularity of mind. Harvard, Mass.: MIT Press.

Flower, L. & Hayes, J.R. (1980). The cognition of discovery: Defining a rhetorical problem. College Composition and Communication 31(1), 229-243.

Gardner, H. (1983). Frames of mind: The theory of multiple intelligences. New York: Basic Books.

Gaskill, W.H. (1986). Revising in Spanish and English as a second language. Unpublished doctoral dissertation, University of California.

Johnson-Laird, P. (1983). Mental models. Harvard, Mass.: Harvard University Press.

Jones, C.S. & Tetroe, J. (1987). Composing in a second language. In A. Matsuhashi (Ed.) Writing in real time: Modeling the production processes. Norwood, N.J.: Ablex.

Krashen, S. (1982). Principles and practice in second language acquisition. Oxford: Pergamon.

Lipski, J.M. (1978). Code-switching and the problem of bilingual competence. In M. Paradis (Ed.) Aspects of bilingualism. Columbia, S.C.: Hornbeam.

McLaughlin, B. (1987). Theories of second language learning. London: Edward Arnold.

Scardamalia, M. & Paris, P. (1985). The function of explicit discourse knowledge in the development of text representations and composing strategies. Cognition and Instruction 2(1), 1-39.

Seliger, H. (1983). The language learner as linguist: Of metaphors and realities. Applied Linguistics 4(3), 179-191.

Without requisite levels of literate knowledge, people may not be able to engage effectively in this kind of integrative learning, as research on minority-language children by Cummins (1984) suggests.

REFERENCES

Anderson, J.R. (1987). Methodologies for studying human knowledge. Behavioral and Brain Sciences 10(3), 467-477.

Arndt, V. (1987). Six writers in search of texts: A protocol-based study of L1 and L2 writing. ELT Journal 41(4), 257-267.

de Beaugrande, R. (1984). Text production: Toward a science of composition. Norwood, N.J.: Ablex.

Bereiter, C. & Scardamalia, M. (1987). The psychology of written composition. Norwood, N.J.: Ablex.

Brown, A., Palincsar, A.S., & Armbruster, B.B. (1984). Instructing comprehension-fostering activities in interactive learning situations. In H. Mandl, N. Stein & T. Trabasso (Eds.) Learning and comprehension of text. Hillsdale, N.J.: Erlbaum.

Butler-Nalin, K. (1983). Revising patterns in students' writing. In A. Applebee (Ed.) Contexts for learning to write. Norwood, N.J.: Ablex.

Case, R. (1985). Intellectual development: Birth to adulthood. Orlando: Academic.

Carroll, J.B. (1985). Second language abilities. In R.J. Sternberg (Ed.) Human abilities: An information-processing approach. New York: W.H. Freeman.

Chomsky, N. (1988). Language and problems of knowledge. Harvard, Mass.: MIT Press.

Cumming, A.H. (1986). Intentional learning as a principle in ESL writing instruction: A case study. TESL Canada Journal, special issue 1, P. Lightbown & S. Firth (Eds.) 69-83.

Cumming, A.H. (1988). Writing expertise and second language proficiency in ESL writing performance. Unpublished doctoral dissertation, University of Toronto.

Cummins, J. (1984). Bilingualism and special education: Issues in assessment and pedagogy. Clevedon, Avon: Multilingual Matters.

The Minimal Chain Principle: A Cross-linguistic Study of Syntactic Parsing Strategies

Marica De Vincenzi
Department of Psychology/ University of Massachusetts

The experimental work presented here is part of a series of studies done on sentence processing in Italian. The goal is to provide a cross-linguistic testing of sentence processing models. Most studies in sentence processing have been done in English. Cross-linguistic study is motivated by the possibility that the processing principles identified to date and consequent processing models, may have been biased by language-specific aspects of English. From this last perspective the comparison of English and Italian is particularly interesting because the two languages differ in the setting of a linguistic parameter. Some natural languages allow phonetically null subjects in tensed clauses (henceforth 'pro', a short form for lexically null '<u>pro</u>nominal' element), while others do not. The two types are instantiated by Italian and English, as in (1) (The * means that the English sentence corresponding to the Italian one is ungrammatical. I give the grammatical English translation in parenthesis). It has been shown that other properties systematically correlate with the null subject properties, including a free process of subject inversion (2).

(1) a. pro CORRERA'
b. *pro WILL RUN.
(he/she/it will run)
(2) a. pro_i CORRERA' $GIANNI_i$
b. *pro_i WILL RUN $GIANNI_i$.
(Gianni will run)

As illustrated in the examples (1) and (2) in Italian there are two types of lexically empty categories that are missing in English: the 'pro' subject in (1) and the 'pro' subject in (2) that is linked to another NP in the same sentence in a filler-gap relationship. (In the standard analysis the post verbal subject is moved form the original pre-verbal position in (2)). The work presented here tests whether these empty elements are processed following the processing principles already identified in English (Crain & Fodor, 1985; Stowe, 1986; Frazier & D'Arcais, submitted; Clifton & Frazier, in press).

SPECIFIC HYPOTHESIS TESTED

The specific processing principle that is tested here uses the notion of chain. A chain is defined as a set of coindexed elements that share one thematic role (e.g., agent, patient, etc.) and one case (e.g., nominative, accusative, etc). The shortest chain is one in which an element is in its unmoved position and it is not linked to any other elements in the sentence, as the NP 'John' in (3):

(3) John runs.

Chains can also have two elements as in the filler-gap relationships of (4) and (5) ('e' designates a phonetically empty structural position):

(4) What $girl_i$ did you call e_i?
(5) $Mary_i$ decided e_i to go.

In (4) the semantic role of the Wh-word (the filler) depends on structural properties of the sentence that follows, namely that there is a 'gap', or empty position in the phrase structure at the direct object position resulting in an empty patient role. Similarly, the

interpretation of the lexically empty structural position (the gap) of the subject of the infinitival 'to go" in (5) is controlled in interpretation by the NP 'Mary' (the filler). In both of these cases, then, there is a chain of length 2 between two elements. Going back to the Italian examples, in (2) there is a chain of length 2 between the Inverted-subject (henceforth, I-subject) and the pre-verbal subject position. In (1) there is a chain of length 1. The lexical empty element in subject position is unmoved and not linked to any other element.

The processing principle tested here is the Minimal Chain Principle (henceforth, MCP):

Minimal Chain Principle:
Avoid postulating unnecessary chain members at S-structure, but do not delay required chain members.

The second clause of the principle amounts to a statement of the "gap-as-first-resort" principle (Fodor, 1978) or the Active Filler Hypothesis (Frazier & D'Arcais, submitted). Here I will focus on the first clause of the principle: it suggests that when a parse is ambiguous between a shorter and a longer chain, the parser will prefer the shorter chain. In the limit, this means that the parser will prefer to have no multiple-element chains at all. Let us take as examples sentences (1) and (2). When a reader gets the verb, she/he knows that there is a lexically empty subject position. But the parse of the pre-verbal subject position is ambiguous between a pro in a singleton chain (as in (1)) or a pro in a multiple-element chain (as in (2)). Several interesting questions arise at this point: does the parser commit itself to a choice of of empty category in the absence of disambiguating information? And if yes, what is the choice that the parser makes? I assume that the parser commits itself on-line to a structural choice without waiting for disambiguation (see Frazier & Rayner,1982) and I will test whether the human parser follows the MCP principle in making choices. This predicts that the parser will initially pursue the singleton chain analysis. This will prove to be correct in (1), but not in (2) where a post-verbal NP occurs. Therefore sentences like (2) should be harder to parse than sentences like (1).

Here I will present two experiments that have tested the MCP principle. The first experiment tested the prediction of the MCP sketched above, using sentences like (6) which are structurally ambiguous between the interpretation (6a) and (6b) because the verb is optionally transitive:

(6) HA CHIAMATO GIOVANNI.
  a. *pro CALLED GIOVANNI.
    (he/she/it called Giovanni)
  b. *pro_i CALLED GIOVANNI.
    (Giovanni called)

If the parser obeys the MCP principle, then the interpretation (6a) should be preferred over (6b). Therefore we predict that when disambiguating material occurs later in the sentence, (6b) should be harder to process than (6a) because a revision of the initial assignment is needed.

The same prediction was tested in a second experiment with intransitive verbs belonging to two different classes, ergatives and unergatives. These two verb classes differ in having the subject originating in post-verbal position (7a) or in pre-verbal position (7b) (Belletti, 1988), and are easily identified because they select different auxiliary verbs:

(7) a. E' ARRIVATA pro.
    * IS ARRIVED pro.
    (she/it arrived)

b. pro HA ESITATO.
*pro HAS HESITATED.
(he/she/it has hesitated)

In the case of a lexically expressed subject in post-verbal position, we have a singleton chain in (8a) and a multiple-element chain in (8b):

(8) a. E' ARRIVATA UNA AMICA.
* IS ARRIVED A FRIEND
(A friend arrived)
b. pro_i HA ESITATO UNA $AMICA_i$.
*pro_i HAS HESITATED A $FRIEND_i$.
(A friend has hesitated)

The MCP principle predicts that a structure like (8a) should be easier to process than (8b) because in the latter case the reader must form a chain relating the post-verbal subject to the pre-verbal position. However, if the verb is ergative, as in (8a) no chain will need to be formed.

EXPERIMENTS

Experiment 1 used 20 triplets of sentences with a temporary ambiguity between a pro-subject (as in (6a)) or an Inverted-subject (as in (6b)): Two sentences had an optionally transitive verb (condition 1 and 2) and one had an obligatorily intransitive verb (condition 3). Each continued with a phrase that disambiguated it. Sample sentences for Experiment 1 are the following:

cond.1: Ieri mattina ha chiamato il capoufficio per chiedergli un aumento di stipendio. (*Yesterday called the boss to ask-him a raise)

cond.2: Ieri mattina ha chiamato il capoufficio per offrirci un aumento di stipendio.(*Yesterday called the boss to offer-us a raise)

cond.3: Ieri mattina ha telefonato il capoufficio per offrirci un aumento di stipendio. (*Yesterday telephoned the boss to offer-us a raise).

The rationale clause in the first example pragmatically forces a reading in which "the boss" is the object of "called" and the matrix sentence has a null subject in pre-verbal position (pro). The rationale clause in the second sentence forces the reading in which "the boss" is the subject, occurring in post-verbal position. In the first case, no chain is needed. In the second case, a 2-element chain must be created when the disambiguating phrase is encountered. Following the MCP the disambiguating phrase in condition 1 will not require reanalysis and will be read quickly. The disambiguating phrase in condition 2 does require reanalysis and will be read slowly. When the verb is obligatorily intransitive (cond. 3: "telefonare" is intransitive in Italian) there is no ambiguity of the post-verbal NP. If the parser is sensitive to this fact, it will not initially choose the pro-subject interpretation, and thus it will not have to perform reanalysis when the rationale clause beginning "per offrirci" arrives.

Experiment 2 used 20 pairs of sentences with ergative and unergative verbs that have post-verbal subjects, as in (8). They occurred in two conditions, with ergative and unergative verbs:

cond.1: Questa volta e' venuta una cara amica ad aiutarci a traslocare. (*This time came a dear friend to help us move)

cond.2 Questa volta ha esitato una cara amica ad aiutarci a traslocare. (*This time hesitated a dear friend to help us move).

Reading Time should be fast for the ergative post-verbal subject sentences (cond.1) as compared to the

Table 1

Average Reading Time for Correct Responses in Experiment 1.

segment #	1	2	3	4	5	%
condition 1 (V_t, pro subject)	1006	1102	950	798	793	.662
condition 2 (V_t, I-subject)	1029	1094	1047	778	853	.718
condition 3 (V_i, I-subject)	991	1124	957	761	812	.929

Note: V_t = verb transitive; V_i = verb intransitive.
*: only in condition 1 the correct response was: "I do not know"

examples: segment #

	1	2	3	4	5
cond. 1	Yesterday/	called the boss/	to ask him/	a raise/	in stipend.
cond. 2	Yesterday/	called the boss/	to offer us/	a raise/	in stipend.
cond. 3	Yesterday/	telephoned the boss/	to offer us/	a raise/	in stipend.

Table 2

Average Reading Time for each segment in Experiment 2.

segment #	1	2	3	4	%
condition 1 (V_e)	989	1043	894	893	.980
condition 2 (V_u)	998	1300	925	881	.883

Note: V_e = verb ergative; V_u = verb unergative.

examples: segment #

	1	2	3	4
cond. 1:	Yesterday/	came a friend /	to help us/	move.
cond. 2:	Yesterday/	hesitated a friend/	to help us/	move.

unergative post-verbal subject sentences (cond.2), because of the necessity of forming a chain in the unergative but not the ergative case.

PROCEDURE AND DESIGN

The 2 experiments were run together. They used phrase-by-phrase self-paced reading task. Subjects pushed a key with the right hand to start each trial, and they pushed the same key to see each successive phrase of the sentence. Only one phrase was visible at a time. When each phrase appeared, the previous one (to the left) disappeared. Subjects were instructed to read at a natural rate, while

maintaining good comprehension. After each sentence, a comprehension question appeared on the screen all at once and the subject had to respond to the question by pressing a button. The question queried the subject of the sentence (for condition 1 and 2 of both experiments) or some randomly chosen aspect of the sentence (condition 3 of experiment 1 and fillers).Reaction Time and question-answering time recorded.

The subjects were sixty college students of the University of Padova (Italy). They received written instructions, then saw 20 practice trials, followed by the experimental and filler trials in randomized order. No subject saw more than one version of any sentence and, accross subjects, every experimental sentence appeared in each condition.

RESULTS

Experiment 1: the data for Experiment 1 are presented in Table 1; the mean reaction times were computed for each segment, after eliminating times that were longer than 5000 msec. and shorter than 100 msec. Reading times associated with erroneous answers to the questions were discarded and the data were normalized by eliminating those subjects (nine) who made over 80% errors. In the critical disambiguating region (segment 3), there was a significant difference in reading time in three conditions, (F_1 (1,50)=3.16, $p<.04$) and (F_2 (1,19) =3.45, $p<.04$). Pairwise comparisons showed that condition 1 was faster than condition 2 (F_1 (1,50)=3.89, $p<.05$) and (F_2 (1,19) = 4.67, $p<.04$) as was condition 3 (F_1 (1,50)=3.75, $p<.05$), (F_2 (1,19)=5.20, $p<.03$). The difference between condition 1 and condition 2 remained significant ($p<.06$) when a regression analysis was used to adjust for length and frequency differences between the verbs. On the 5th segment there was still a significant difference in reading time (F_1 (1,50)=,2.92 $p<.05$), (F_2 (1,19) =2.97, $p<.06$). A pairwise comparison showed that the only significant difference was between condition 1 and 2 (F_1 (1,50)=4.90, $p<.02$), (F_2 (1,19)=4.65, $p<.04$). The questions after the sentences were answered more accurately in condition 3 than in conditions 1 and 2, (F_1 (1,50)=46.62, $p<.001$), while the latters did not differ significantly from one another (F_1 (1,50)=2.68, $p<.11$).

Experiment 2: the 2nd segment, containing the verb plus the post-verbal subject, was read faster for ergative than for unergative verbs. The difference was significant both in a simple analysis of variance (F_1 (1,59)=52.28, $p<.001$), (F_2 (1,19)=72.39, $p<.001$) and in a regression analysis, adjusting for length and frequency differences among the verbs ($p<.04$). Similarly, questions were answered more accurately following ergative sentences than following unergatives (F_1 (1,59)=13.31, $p<.001$.).

DISCUSSION

Both experiments confirm the hypothesis that the parser follows MCP, preferring shorter chains. In experiment 1 the sentences whose pragmatics are consistent with the theoretically preferred pro-subject analysis are easier to read than sentences whose pragmatics force the theoretically unpreferred Inverted-subject reading. This result is especially surprising because a pro has a contextually given antecedent in normal usage, but it does not in the experimental setting. In context, therefore, there should be an even stronger preference than the one found here. When the sentence contains an obligatorily intransitive verb (as in condition 3) eliminating the ambiguity of the post-verbal NP,

reading time for the critical segment "per offrirci" is fast relative to "per chiedergli", as if it does not occasion any reanalysis.

Experiment 2 shows that sentences with a subject after an ergative verb are easier to process than similar sentences with unergative verbs. This confirms the MCP principle that the parser prefers analysis that require chains of the smallest possible length. It also provides support for an analysis of the syntax of Italian ergative verbs like that presented by Belletti (1988).

CONCLUSION

We have shown that the human sentence parser follows a structural principle of Minimal Length of Chain in parsing lexically empty elements and in establishing filler-gap relationships. This principle can be seen as a generalization of the Active Filler principle (assign an identified filler as soon as possible) which has received support from experiments done on English and Dutch (Clifton & Frazier, in press; Frazier & D'Arcais, submitted). Thus, we may offer a uniform account of processing empty categories which holds for languages varying in their inventory of empty categories.

This principle, finally, is consistent with a view of language processing as operating under time pressure and short-term memory limitations. One can assume that constructing a multiple-element chain is costly in terms of time, because there is a delay in assigning an element to its argument position. Keeping elements in memory in an unstructured form is generally assumed to be more costlythan mantaining them in an unstructured form (Miller, 1965). Similarly it can be argued that a multiple-element chain is costly in terms of effort, when several elements must be kept in short-term memory (Wanner & Maratsos, 1978). The consequence is that chians containing the smallest possible number of elements will be constructed before chians involving more elements because the former are faster and place less burden on the language processor.

REFERENCES

Belletti, A. (1988). The case of unaccusatives. Linguistic Inquiry, 19, 1, 1-34.

Clifton, C., Jr. & Frazier, L. (in press) Comprehending sentences with long distance dependencies. In M. Tanenhaus & G. Carlson (Eds) Linguistic Structure in Language Processing. Dordrecht: Reidel.

Crain, S. & Fodor, J. (1985). How can grammar help parser? In D. Dowty, L. Kartunnen & A. Zwicky (eds), Theoretical Perspectives on Natural Language Parsing. Cambridge: Cambridge University Press.

Fodor, J. (1978). Parsing strategies and constraints on transformation. Linguistic Inquiry, 9, 427-474.

Frazier, L. & Rayner, K. (1982). Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences. Cognitive Psychology, 14, 178-210.

Frazier, L. & Flores D'Arcais, G. B. (submitted). Filler-driven parsing: A study of gap-filling in Dutch.

Miller, G. (1965). The magical number seven, plus or minus two, or some limits on our capacity for processing information. Psychological Review, 63, 81-96.

Stowe, L. (1986). Parsing Wh-constructions: evidence for on-line

gap location. Language and Cognitive Processes, 1, 227-245.

Wanner, E. & Maratsos, M. (1978). An ATN approach to comprehension. In M. Halle, J. Bresnan & G. A. Miller (Eds) Linguistic Theory and Psychological Reality. Cambridge, Mass: MIT Press.

MULTIPLE CHARACTER RECOGNITION - A SIMULATION MODEL

Sebastian Koebe & Gerhard Deffner[1]

University of Hamburg

In this article we describe a model which simulates the process of human recognition of handwritten characters. For pragmatic reasons, our first attempt is limited to the subset of the 10 digits. Work on this problem is not new, however, and the main requirements of successful recognition are well known: a model should account for the human ability to recognize characters:

(1) in all possible positions in a given display
(2) in a display containing multiple objects (numbers)
(3) of different sizes
(4) of varying shape and form
(5) if they are distorted (discontinuities of lines)
(6) in any orientation
(7) if they overlap

Our model can cope with problem 1 through 4, and 5 to some extent. It is a hybrid system in which the use of serial vs. parallel processes is contingent upon assumptions based on empirical data. In general, it can be described as an early selection system, recognizing numbers in a serial order, after parallel information about the whole visual field is utilized early in the process. This type of capacity limited recognition model first uses positional information about objects in the input as the basis for further object selection, then attention is focused on single items to perform the computationally expensive process of recognition.

The main idea is that requirements 1-3 are dealt with by a process of selecting and standardizing individual objects from the display. Recognition is achieved through a PDP-network. These procedures are called iteratively until all objects are recognized.

The input to the system is a 400 x 200 pixel array produced with a paint program tool. This matrix of binary units stands

---

[1] We want to thank the Fulbright Commission for providing a grant to the first author to spend a year at the Cognitive Science Institute at LaJolla and David Rumelhart who was supervisor during that year and who provided for ideas and encouragement to enter the PDP-paradigm.

for the output of the foveal part of the retina covering the centre of the visual field. Although the output from receptors in the retina is frequency-modulated with respect to the intensity of a stimulus (behavior over time), we assume a static pattern. The matrix of binary units can be thought of as a 'snapshot' of the firing pattern in the retina containing the relevant information for the process of pattern recognition. Another reason for working with a static array is the fact that only such information from the visual field is processed semantically which comes from the periods of relative rest of the eye during fixations and not from saccadic eye-movements.

The fact that the information processing capabilitity of the visual system is different for syntactic and semantic information (c.f. Rayner, 1975) has motivated our choice of two mechanisms in the model. The performance of the first is similar to the recognition of syntactic information by the human visual system. In the same vein that features such as word length or size of letters can be made out at some distance from the fixated point, we assume that objects can be picked out from the array we use. After this selection, another mechanism takes over - that of character recognition.

1. Component processes of the model

1.1. Deriving a positional map

The first step in analyzing a given binary array is to determine the number of objects and their position in the input. This is done by a fast parallel algorithm (assuming parallel hardware as in the brain), which computes the center of gravity for each (potential) object in the display. Consider two fully connected layers of units, with the first layer being the retina receptors and the second layer being a one-to-one map of the first layer with the same number of units. Every single receptor in the first layer, which can have only one of two possible states ('on' or 'off'), is connected to every unit in the second layer (Figure 1). When a receptor is active ('on' in the binary array), it sends activation to all of the units in the second layer according to the weights of the connections. The weight between a unit in the first layer and all other units in the second layer is set according to the Gaussian function of the distance between the units. Every unit in the second layer receives weighted input from all units in the first layer. All incoming activation is summed, thus yielding a total activation value for each unit in the second layer. This array of total activation can be illustrated as a map of activation values as in Figure 2. The local maxima in the landscape (the hilltops) represent the centers of gravity of objects in

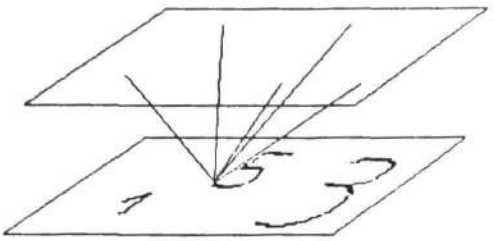


Figure 1: Weighted Connections

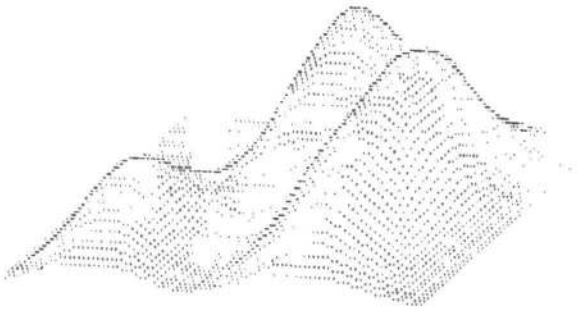


Figure 2: Activation landscape

the input, the number of local maxima is equal to the total number of objects. X- and Y-coordinates of these provide positional information for the next steps. By means of an attentional mechanism one of these is selected for further analysis, namely their recognition.

1.2. Standardisation of selected objects

Next, the object associated with the selected center of gravity has to be isolated in the input array. Given the position of the center of gravity for an object, all the adjacent 'on' units surrounding the center will constitute the isolated object. The criterion for adjacency in our model is very strict. Adjacency is at first defined as a distance of one unit. This means that only objects which are built out of directly adjacent active units are isolated as one. Other adjacency criteria can be chosen to allow for discontinuity of lines. Information about the height of local maxima can also be used to adjust the criterion. At present, our model uses a static criterion, however.

Once an object is isolated, its size can be determined as the maximum number of pixels horizontally and vertically. Next, the object is mapped onto a square matrix (see Figure 3a). The resolution of this matrix is then reduced to yield a standardised 10 by 10 bit matrix (Figure 3b). Arriving at such a final matrix, we have satisfied the above requirements concerning position and size.

1.3. Character recognition

On an abstract level, this task can be described as that of detecting <u>inter</u>-character variation in the face of <u>intra</u>-character variation (different shapes of the same digits). This can be thought of as a mapping problem. All input matrices

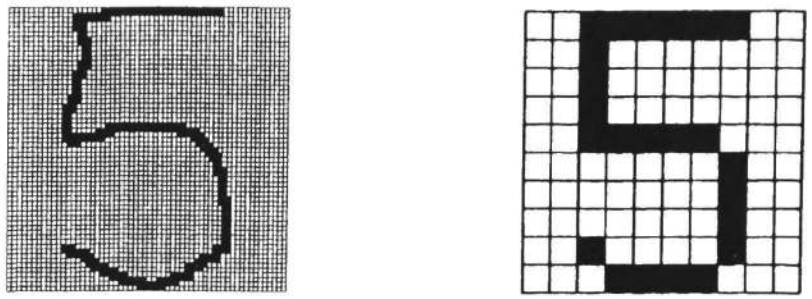


Figure 3: Selection at original (a) and reduced resolution (3)

containing the same digit (for instance "5") should be mapped onto the same output. Thus, there are 10 possible output states. The question is, what the representational form of the output should be. On the one extreme would be a relatively dense representation using 10 units, one of which is "on" for each digit ("grandmother cells"); on the other extreme would be 10 patterns of the same size as the input (10 x 10 units), each of which represents a prototypical digit. Dense discrete information is what we require as output from a recognition mechanism. At an intermediate level, however, the 10 x 10 representation is more plausible: for one, the number of possible output patterns is kept large, thus imposing no limits on the recognition capacity. Also, we want to separate a final decision phase (Sternberg, 1969) from the earlier process dealing with variations of shape that results in differential information about the presence of various features in the input array. In this way, a level is provided where featural (bottom-up) information can be combined with contextual information prior to the final decision about the identity of the stimulus (allowing for contextual effects like the Stroop phenomenon or context enhancment in reading). This distinct level is a prerequisite of more comprehensive models of human recognition as it is found for example in Rumelhart and McClelland's Interactive Activation Model of Context Effects in Letter Perception (1981).

Traditionally, shape variance has been dealt with through mechanisms of feature analysis (Selfridge & Neisser, 1960). A major problem has been that a comprehensive set of features must be defined by the designer of the system for each character set to be recognized. As a more flexible alternative, connectionist networks can be used which 'learn' features from material presented to them and then generalize to new input. After sufficient learning, the knowledge about relevant

features is embedded in the hidden layer of such networks and can be utilized in mapping the input matrix onto the 10 x 10 matrix of the intermediate level.

The connectionist network we use has three layers and employs backpropagation as the learning procedure (Rumelhart, Hinton & Williams, 1986). It consists of 100 input units, 50 hidden units and 100 output units with no direct connections from input to output. During learning, input patterns in the form of 100-element vectors (representing the 10 x 10 matrix) are mapped onto corresponding prototypical target vectors (standing for the ten prototypical digits). Since it is not practical to map all possible permutations ($2^{100} = 1.2676506 * 10^{30}$) of the input vector space on to the 10 target vectors, we took a sample of the input vector space, by asking 20 subjects to write digits on a computer screen. The same standardisation procedure as described earlier was used to transform their handwritten digits into the 10 by 10 standard form. In order to enlarge sample variability, white noise was added to the input vectors. A total of 1000 input vectors was used for learning.

After learning, the network can now be used to produce 100-element output vectors for any new input vector. The elements of the output vector can assume continous values between 0.0 and 1.0, thus reflecting the degree to which certain features are contained in the input material.

The decision which completes recognition is accomplished by relating this vector of continous values to the 10 prototypical binary vectors. Similarities between continous and prototypical vectors are computed and the name of that protoype is used as a label for the item to be recognized, for which similarity is greatest and above a preset threshold. If all similarities are below threshold, a decision of "no known character" is made. These labels provide the desired discrete output states.

2. Putting it all together

The interaction of the various mechanisms is illustrated in Figure 4. The 400 x 200 input bit matrix (1) is made available as a positional map (2) also. The main control structure is that of a loop in which Attention selects one position at a time and feeds coordinates (3) to Object Standardisation. Object Standardisation isolates the corresponding object from the input matrix and transforms it to the standard 10 x 10 matrix (4). The matrix is then fed to Character Recognition which outputs discrete characters. This is repeated (the digit "5" is used as an example of the first cycle) until the positional map is exhausted.

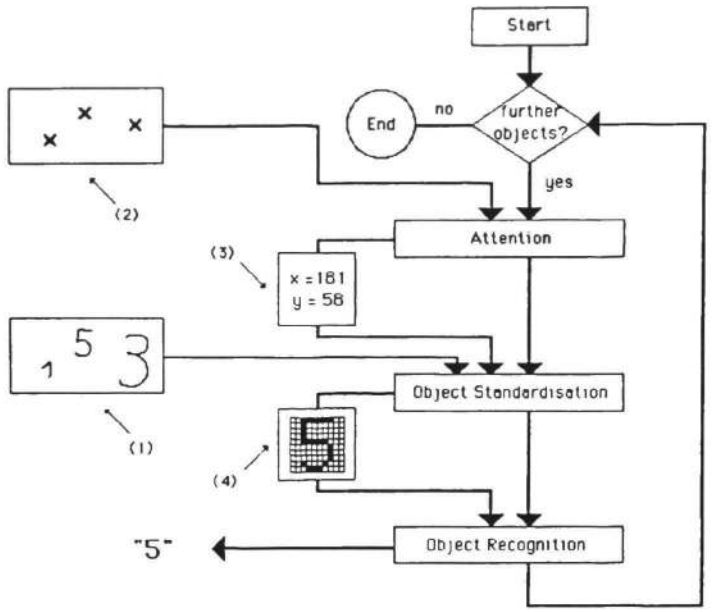


Figure 4: Interaction of the components and flow of control

3. Discussion

The main characteristics of the process of character recognition used in our model are: First, the limitation of recognition capacity available at a time (there are only 100 input units to the PDP-network). This limitation, imposing the requirement of an attentional mechanism for focussing on particular areas in the input, appears justified when we consider the enormous size of the initial input vector space of the 400 x 200 pixel matrix. Mappings from a space of 2^{80000} input vectors seem out of the question. Due to the selection of single items and subsequent reduction in resolution, the number of input units necessary for recognition can be reduced dramatically. The decision in favour of this practically inspired approach takes into account the trade-off between parallel processing capacity and time (fast parallel vs. slow serial recognition).

Second, we asssume automatic parallel processes providing information for the process of selecting characters from the visual field. This is the basis for early selection. Our model only computes a positional map of objects, but empirical data suggest that the human visual system also provides information like for instance colour and texture maps of the primary input. In an elaborate system, an attentional mechanism (which we did not detail) would have to integrate all such sources of information in order to control the selection of items in the input.

Third, in this model variance in the input is reduced stepwise. Before a featural analysis of characters is performed, position and size variations are eliminated. With regard to the recognition problems 6 and 7 (overlap and rotation of characters) the presented approach does not suggest obvious solutions.

In principle, the model can be extended to cover larger character sets. We would have to train the PDP-network on examples of new characters in order to generalize to the relevant structures[2]. It is interesting to note that the system learns through positive examples only. Information from the vast space of vectors not representing characters would not allow for a detection of regularities. The variation resulting from random sampling of this space is huge and unsystematic in relation to the variance of the extremely small percentage of vectors which do represent character.

References

Rayner. K.E. (1975) The perceptual span and peripheral cues in reading. Cognitive Psychology, 7, 65-81.

Rumelhart, D.E. & McClelland, J.L. (1981) An interactive activation model of context effects in letter perception: Part 2. The contextual enhancement effect an some tests and extensions of the model. Psychological Review, 89, 60-94.

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, Volume 1. Cambridge: MIT Press.

Selfridge, O.G. & Neisser, U. (1960) Pattern recognition by machine. Scientific American, 203, 60-68.

Sternberg, S. (1969) The discovery of processing stages: Extensions of Donders' method. In: W.G. Koster (ed.) Attention and Performance II. Amsterdam: North Holland.

---

[2] The resolution of a 10 x 10 standard matrix would probably be too small to represent letters with all important features. This could be overcome by increasing the size of the matrix.

NETZSPRECH - Another Case for Distributed 'Rule' Systems

Georg Dorffner
Dept.of Medical Cybernetics and Artificial Intelligence
University of Vienna, Austria
and: Austrian Research Institute for Artificial Intelligence

Abstract: This paper compares conventional symbolic rule systems with distributed network models, considerably arguing for the latter. NETZSPRECH - a network that transcribes German texts similar to NetTalk is first introduced for this purpose and serves as an example for the arguments.

1. Introduction

Models in artificial intelligence (ai) and cognitive science rely mostly on the assumption that cognitive systems (such as the human brain) cannot only be described but also efficiently modeled using symbolic descriptions and rules to combine them. However, many aspects of human perception and cognition are left out by those systems, which can be attributed to a great extend to the limitations of symbol-and-rule systems due to their brittleness. Recently, research on parallel distributed processing (PDP) models has shown that those models can have the power to overcome some of the said limitations. Nevertheless, there still exits many critics and sceptics toward the PDP paradigm (such as in Pinker&Prince 1987) who neglect the points where PDP models can be superior.

The purpose of this paper is to describe Netzsprech, a new implementation of the Nettalk model by Sejnowski&Rosenberg (1986) adapting it to the German language, and to use this model to argue for PDP models and against models merely based on symbols and rules. Netzsprech is a network that (similarly to Nettalk) learns to translate text into a phonetic transcription from examples, i.e. it learns to pronounce German words. Reading and pronouncing a script is a process that appears suitable to show the advantages of a distributed model over a symbolic one.

2. The Netzsprech model

Netzsprech is a basic so-called associative network (AN) (Rumelhart, McClelland 1986) consisting of three layers of units. Many of the processes that can be modeled by an AN appear to be rule-governed in that symbolic rules can describe the association to a large extent. Thus, ANs can be viewed as distributed 'rule' application systems. I put 'rule' under quotes here, because it has to be understood in a much more relaxed sense than rules in conventional AI systems. Distributed 'rule' application can overcome many of the restrictions symbolic rule application is bound to. The architecture of Netzsprech is depicted in Fig.1.

A perfect example of seemingly rule-governed behaviour with

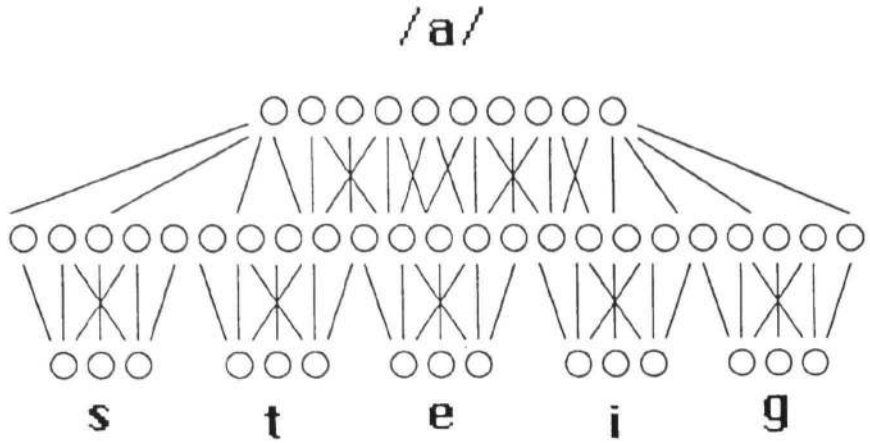


Fig.1

nevertheless a lot of exceptions is the human ability to read text aloud, that is, to transform the written representation of a text to the phonemic pattern that is to be spoken. T.Sejnowski and R.Rosenberg (1986) have shown in an impressive manner that their NetTalk model could learn to transcribe English text into phonemes, which were then fed into a phonemic speech synthesizer. Without doubt, English is the Indoeuropean language with the most discrepancies between graphemic and phonemic patterns. Nevertheless, NetTalk could learn to pronounce simple English text with an error rate well below 10%.

This paper describes a network that learns the pronounciation of German words in a very similar fashion to NetTalk. Hence, I called it 'Netzsprech'. With this implementation I want to demonstrate the advantages of distributed learning and representation schemes which led me to believe that those schemes will be a necessary complement for AI programs in the future.

3. What Netzsprech does

The input layer (fig.1) is divided into five clusters which encode five letters of German text. The letter in the middle is the one to be transcribed, the two on the left and on the right, respectively, serve as context information to aid the transcription. Sejnowski and Rosenberg (1986) used seven letters (i.e. a context of three letters on each side), but as it turns out, in German generally a two-letter context is sufficient, if one excludes difficult foreign words.

The representation of letters in the five input clusters is a local one, that is, one unit in each cluster, when activated, corresponds to exactly one letter in the text. As a result of this, only one unit can be active per cluster at any given time. As there are 31 letters (including umlauts, the 'scharfe s', and a space), the input layer consists of 5 times 31 units.

For representing (encoding) the phonemes in the output layer a

different strategy was used. Here one expects to find similar codes for similar phonemes. One example in German is the letter 'd' that can be pronounced as a [d] (in a Wort like 'der') or as a [t] (at the end of a word like 'und', this is known as final devoicing in German). Although there are two different phonemes for the letter 'd', these phonemes are nonetheless very similar to each other (in fact, they only differ in their voicedness). For the model to be adequate, these two similar phonemes should be encoded in two similar output activation patterns. This was done using a binary code distributed over the activations of 10 output units.

The hidden layer which serves to provide the possibility to learn arbitrary pattern mappings between input and output, consists of 30 units.

4. The training phase

The training set consisted of a list of the 1000 most common words in German (according to Meier (1978)) plus their transcription. The transcription that was available in machine-readable form (produced by a rule-based model designed by Pounder and Kommenda (1986)) had to be hand-edited to take account of the fact that the Netzsprech network requires a one-to-one mapping between letter and phoneme. Thus, when there were cases where a whole sequence of letters has to be pronounced as a single phoneme, the phoneme was taken as the transcription of the first letter. The other letters were transcribed as 'silent' (using a pseudo-phoneme [+]). For example, the sequence /sch/ has to be pronounced as [ ] in most cases, so that the transcription of the word 'asche' became [a ++$] (where [$] is the schwa-phoneme). The reverse case - one letter has to be pronounced as a sequence of phonemes - does not occur in standard German, if one writes affricates (like [ts] for the letter /z/) as one phoneme.

Training consisted of taking each word from the list (in the order they appear) and its transcription and presenting it to the network, letter by letter, encoding both letters and phonemes. After each presentation (5 letters plus a phoneme) the delta rule was applied and the weights were adjusted.

5. The results

Fig.2 shows the learning curve of the network. The x-axis shows the number of words used for training and the y-axis the number of correctly transcribed letters and words. Presentation of 3000 to 4000 words (that means, the list of 1000 words was scanned 3 to 4 times) is sufficient to bring the network very close to convergence, with the error rate for letters going down to less than 10 %. Most of the errors that remain at this point are minor, i.e. only 1 to 2 of the 10 featurse are produced incorrectly. The understandability (the phonemes were subsequently synthesized) is already extremely high.

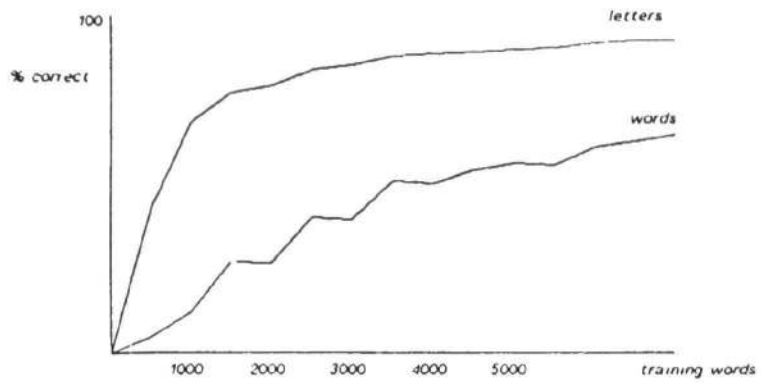


Fig.2

6. Discussion of the network behaviour

At no point in this paper do I want to claim that the model I am describing is a valid model for the complex process of learning how to talk (neither did Sejnowski&Rosenberg). Nevertheless, the model shows some very important behaviour that - if conceived as part of a much more complex model - seems to reproduce many aspects of human processes of understanding and learning in a very plausible manner.

The problem Netzsprech was faced with is one that, to a large extent, is describable with a symbolic rule system, containing rules like

(1) #s{p,t} -> [ ] (2) s -> [s]

meaning that 's' is to be pronounced as an [s] by default and as an [ ] at the beginning of a word (marked by '#') in front of a 'p' or 't'. However, as I am going to show, the distributed system in Netzsprech is superior to the rule model in that it corresponds more closely to human processes and has the advantage of being easily learnable and adaptable.

6.1 Rules are applied simultaneously

One of the weak points of symbolic rule systems is that they almost always involve search or other serial processes. For example, if there exist pronounciation rules for every letter in German, then in a first step all rules for the letter currently considered have to be found. In a second step, all rules among them have to be chosen that match the current context ('matching cycle'). Of course, rule matching could be done in parallel, which reduces this search process to one step. But after that, the system most of the time is faced with the problem of conflict resolution, the selection of the rule to apply among several that all match the context. This can be the case with the two rules given above, in a context like '_ # s p i'. The time it takes to solve problems of rule conflict increases when we add more rules, which is counter the experience that humans

use less time with additional knowledge rather than more.

To get around the problem of rule conflict, the rules could be written in such a way that only one rule applies per context. In other words, a rule would have to be written for virtually every possible context. For example, there would have to be a rule for 's' in front of 'p', where it is pronounced as an [ ], but also one for 's' in front of every other consonant, where it is pronounced as an [s]. For the domain being discussed this might not be impossible to do, but problems are conceivable, where the possible space of contexts is so large that it becomes very inefficient if not impossible to provide a rule for every context.

But even if it were possible in every case, one important property of the original rule scheme is lost: One can no longer say, what is the 'default' rule for a given letter, for example, that 's' in the majority of the cases is pronounced as [s] and therefore this output is the most likely one. In the case where we had two (sometimes competing) rules, by virtue of the lowest priority one of the rules was the default. If little or no context was specified, this rule applied and gave the most likely output.

Thus, in rule based systems, there is a trade-off: Either the system can easily generalize and provide default outputs for unknown inputs, but then it shows rather implausible processing time relations. Or the system can produce the output more plausibly, but then the power for generalization and default outputs is lost.

A distributed system like Netzsprech, however, exhibits both advantages at the same time. All the distributed 'rules' are applied in parallel, so there is never anything like conflict resolution, let alone serial matching processes. Also, as the size of the network is fixed, acquiring new knowledge cannot make the process slower. At the same time, the model shows full power of generalisation for inputs it has never seen before with an inherent default mechanism. Consider the following example from Netzsprech: If an 's' is presented to the trained network without any context (i.e. all context units are set to 0, note that this case never occured during learning) the default (or most likely) output, [s], is activated. Where no such default exists, for example for an 'x' (a very rare letter in German proper), the output is unspecified (i.e. some arbitrary output which nevertheless is closest to the most frequent phoneme corresponding to 'x'). If now, in the case of the 's', a context is added that in German should alter the pronounciation (like a '#' in front and a 'p' afterwards), the output in fact changes to (the now default case) [ ]. One can see from these examples, that the model incorporates both default and more specififc rules at the same time. The context changes something in the output only when it, together with the central letter,

constitutes a case for a more specific rule to the default.

6.2 The output has continuous values

A major property of a distributed network is that it allows for activation values along a continuous scale, indicating how strong a specified hypothesis is. Returning to the above example, when 's' is presented alone to Netzsprech, the output is the default, an [s]. However, the strength of the output is lower than it is when 's' is presented in the more specific context 'ase'. The reason is that in the former case, without any further specification, the 's' is pronounced as an [s] only with a certain probability, if the (unknown) context were '#sp', then it could also be a [ ].

This property of the system can be useful when incorporating the network into a bigger one. For example, one could conceive of another component that controls the movements of the articulatory organs. This component would receive input from several components; mainly, of course, from Netzsprech, but other inputs that influence the movement of the articulators would be possible, too. If now the input from Netzsprech has a rather low level, other sub-systems could more easily 'push' the control component into another direction. For example, even when Netzsprech says [s] the actual pronounciation might turn out to be [ ], when the output was a default case. In the more specific case, with higher output activations, this scenario is less likely. Thus, continuous output values are important for modeling contextual and con-situational effects.

In a symbolic rule system, it is much more difficult to obtain such behaviour. Symbolic rules are much more isolated from each other and from the rest of the system. Making them influence each other is much harder and might not even be efficiently possible. The advantage of the network model is that it provides easy entry points - the units and connections themselves - for other system components to modify the behaviour.

6.3 Rules are learned according to environment

Although the behaviour of the network after learning appears to be rule-governed (such as: a feature F in pattern A produces a feature G in pattern B), none of the 'rules' has to be stated explicitely or beforehand. The network can 'find out' the rules (or whatever we want to call them) by itself from the examples, which then cover exactly the cases in the training set and those derived through extrapolation of the rules. In other words: No one has to stipulate the r presented. Furthermore, as Rumelhart (1986) points out, each 'rule' currently incorporated in a PDP model is applied with exactly the strength that corresponds to its fitness.

6.4 Networks are robust

In a symbolic rule system it is crucial that at any point in time all the needed rules exist and are accesible. When one of them is missing, the system might fail completely. In a network this can never happen. As the 'rules' are represented in a highly distributed manner and - in addition - are all learned from the environment, they cannot easily get lost, not even by partial destruction of the system.

7. Conclusion

Much more could be said about distributed 'rule' systems (Dorffner 1987, 1988). However, this short discussion should already have made clear the advantages of such models. The results suggest that to model cognitive processes such as understanding and producing natural languages - which are much more complex than the task of Netzsprech but largely based on similar mechanisms of association - one will have to include PDP or related models in the future.

References

Dorffner G.: NETZSPRECH: A Network Learns German - Description of the Model and Discussion, Tech.Report 87-01, Dept.for Med.Cybernetics and AI, University of Vienna; 1987.

Dorffner G.: Modeling Gestalt Phenomena in a Distributed 'Rule' System, in: Trappl (ed), Cybernetics&Systems '88, Kluwer Academic Publishers; 1988.

Hinton G.E., McClelland J.L., Rumelhart D.E.: Distributed Representations, in Rumelhart & McClelland (1986).

Pinker S., Prince A.: On Language and Connectionism: Analysis of a Parallel Distributed Processing Model of Language Acquistion, Dept.of Brain and Cognitive Science, MIT, occ.paper #33; 1987.

Pounder A., Kommenda M.: Morphological Analysis for a German Text-to-Speech System, in Proceedings of the 11th International Conference on Computational Linguistics (COLING-86), Bonn, FRG; 1986.

Meier H: Deutsche Sprachstatistik, Georg Olms Verlag, Hildesheim, 1978.

Rumelhart D.E., McClelland J.L.: Parallel Distributed Processing, Explorations in the Microstructure of Cognition, Vol 1&2. MIT Press, Cambridge, MA; 1986.

Sejnowski T.J., Rosenberg C.R.: NETtalk: A Parallel Network that Learns to Read Aloud, Johns Hopkins University, Tech.Rep. JHU/EECS-86/01; 1986.

Reiner M. and Finegold M. Technion, Israel Institute of Technology

INTUITIVE NOTIONS OF LIGHT AND LEARNING ABOUT LIGHT

INTRODUCTION

A great deal of research effort has been invested in the identification of naive concepts held by students in different areas of science, and on the impact of science lessons on naive beliefs. According to Posner et al. (1982), the identification of concepts held by students is important since: "...a person's central concepts are the vehicles whereby a given range of phenomena become intelligible. Such concepts can be linked to prior experience, images, or models which make them appear intuitively obvious.."

Disessa (1981) claims that "intuitive physics is a rather well developed and exceedingly robust system that can potentially interfere with 'proper' physics." This implies that intuitive beliefs held by students before learning play a major role in formulating new concepts. Students approach new topics with a framework of related ideas which are derived from past experience. When faced with teachers' statements or notebooks, or with the results of their own investigations which are not in keeping with the conceptual framework they hold, students either modify their own views, or according to Keneth Lovell (1980), maintain two separate ideas about the same concept and apply them to different situations. Lovell claims that one of the two ideas is used for passing exams in schools, and the other, based on intuition, is used for explaining every-day phenomena. A model used by some cognitive scientists fits well with the hypothesis of interaction between the child's different ideas and the manner in which these ideas evolve with teaching. This model is based on the hypothesis that information is stored in the memory in various forms - schemas - and everything we say and do depends on the elements or group of elements of this stored information.

There is considerable research evidence to suggest that the context or the phenomenal setting of a task or problem influences an individual's performance. Driver and Bell (1986), suggest that the learner's interpretation of the task will depend on preexisting notions which arise from experience based intuition as well as on more formally acquired concepts. Studies which have documented children's conceptions have acknowledged the base concepts underlying learning, but have not critically addressed the question of the mechanism by which former knowledge controls the newly acquired knowledge. Piaget investigated the development of children's ideas about natural and mechanical phenomena. He concluded that young children do not explain phenomena on the basis of causal linking but instead attribute intention and desire to the objects themselves. As they mature they gradually outgrow this kind of belief in favour of more causally oriented thinking. Resnick and Chi (NSTA) note that there are two fundamental elements in this theory: constructivism and logical determinism. The latter term refers to the theory that children, while growing up, develop a set of general logical structures necessary to scientific thinking. In this view, misconceptions are a result of not having the logical structures needed for scientific reasoning.

Constructivism, the notion on which this work is based, refers to the idea that people build their knowledge for themselves. Personal knowledge is a complex, incompletely-understood outcome of an ongoing process of construction and interpretation. Thus the meaning of a concept develops as a result of interaction between meaning existing in the learner's mind and reality.

Kelly's notion of "Constructive Alternativism" in personal development is very much in line with constructivism (Pope and Gilbert, 1983). According to his position people understand themselves, and their surroundings, and predict future events, by constructing tentative models and evaluating them according to their personal models. The criteria used to evaluate new models are based on knowledge gained in the past. This knowledge is organized in personal representational models of the world which make sense of events and which are used to describe personal experience, to predict future events and to assess previous predictions. According to Kelly, any event is open to as many reconstructions as our imagination will allow. There is no absolute truth, but just "nuggets of truth", which are tested by their power of explanation and prediction, and replaced by what Disessa calls a more powerful "piece of knowledge". Thus the former knowledge has a crucial role in the mechanism of acquiring new knowledge.

Piaget suggested that two principal mechanisms govern the learning of new concepts: Assimilation - by which a new concept is interpreted in terms of knowledge that has been acquired in the past; and Accomodation - by which existing knowledge is adjusted according to the newly acquired concept. Resnick and Chi (NSTA) suggest that "children's failure to think scientifically comes not from logical disabilities, but from not having acquired key organizing principles for some domain of knowledge".

There is, however, a lack of information about the actual processes which take place when new concepts are learned. Do all former beliefs influence the acquisition of a new concept in the same way? If not, what are the factors related to past knowledge which will influence the meaning of a given concept more than others? Which of the established concepts and relations in the existing schemata are more dominant than others? In what way do they govern the acquisition of new knowledge?

THE FOCUS OF THIS STUDY

The research reported here deals with the identification of naive concepts of light held by students who have never formally learned about light, and with relations between intuitive knowledge acquired by past experience and new knowledge students are required to learn. We focus on knowledge about light for a number of reasons:

1) Light phenomena are all-pervasive. Children experience light as something which hits the eye, even causing physical pain if it is too bright.

2) Common expressions like "seeing the light", "throwing light upon", and the psalmist's "valley of the shadow of death", all raise questions concerning widely held concepts of light.

3) Links may be discerned between the historical development of concepts of light and concepts of light based upon everyday experience. (Pythagorean emission theory of particles bombarding the eye and experience of blinding light; Visual rays projected by the eye at infinite velocity and the simple observation that we see the stars at night immediately upon opening our eyes; the Platonist assumption that sight results from interaction among sunlight, particles emitted by objects seen and the eye, and the everyday experience that a light source, an object to be seen, and an open eye are all necessary for sight).

A PILOT STUDY

Interviews were conducted to examine non scientific beliefs about light held by persons with a good formal background knowledge of light. Amongst the interviewees were students majoring in physics and physics teachers. Questions asked related to everyday phenomena and to topics encountered in formal learning. This was done in an attempt to differentiate between the kinds of knowledge used in explaining phenomena as experienced and phenomena as learned in a classroom. For example, we asked: Why does TV or radio noise often accompany lightning? and, What causes the Crookes radiometer vanes to rotate?

The analysis of interview protocols showed that persons with good physics backgrounds often held beliefs incongruent with the formal science they had presumably learned. Responses suggested that although the questions called for knowledge in the same area, interviewees applied different sources of knowledge in dealing with everyday and with formal science problems. In addition the terminology used for dealing with questions on everyday phenomena often differed from that used in answering questions on formally learned phenomena. Thus in moving from everyday to formal science, terms like vibration, strength, colour, and mixing, gave way to terms like frequency, force, wave length, and superposition.

One concept, that of light particle, was found to be common to both knowledge sources, being seen as a small round material object rather than as the abstract notion encountered in formal learning about light. This suggests that the idea of light as an actual material substance was held by interviewees irrespective of the level of their formal achievement in physics.

In an attempt to identify possible sources of this naive and apparently intuitive conception, a study was initiated of a group of highly achieving 17 year olds who had never learned formally about light, and who planned careers in engineering or science.
Goals of the study were:

1) Identification of naive conceptual frameworks about light;
2) Identification of interrelations between established and newly acquired knowledge;
3) Development and evaluation of a method for restructuring knowledge about light.

THE IDENTIFICATION OF PRESCIENCE CONCEPTS OF LIGHT

Eleventh grade students majoring in the electrical engineering trend in a technological high school were interviewed in order to identify explanatory conceptual frameworks on the nature of light. In order to check for consistency of responses, each phenomenon was discussed in more than one way. Questions took the form:

1) What shines in each of these light sources? (Students were shown and were able to examine: incandescent, fluorescent and neon lamps; a TV screen; a candle;a burning match; and a gas flame).

2) Why do you see different colours when you put different materials in a flame?

3) Why does the light from an ordinary electric light make the Crookes radiometer rotate, but the lab laser light doesn't?

4) You land on an unknown planet lit by a sun radiating red light only. What are the colours visible on that planet?

5) What is a rainbow?

6) A lighted candle is placed on the floor of the room. Draw a line to show the furthest place reached by the light.

7) Is there any place in the room which the candle light doesn't reach? If there is, can you see the candle from such a place?

8) How is it that we have sunlight (twilight) after sunset?

9) A white shirt illuminated by red light appears red, and a red shirt illuminated by white light also appears red. Explain.

10)How does coloured glass change the colour of light passing through it?

It was found possible to classify students' conceptual frameworks with respect to five topics: light sources; propagation; the nature of light; light and sight; light and colours. Student beliefs were grouped for each topic.

The most popular beliefs concerning light sources were that they are all hot bodies, that any material can become a light source if heated to a sufficiently high temperature, and that denser materials require more heat to make them shine. Other popular beliefs were that light sources are created by chemical reactions such as the reaction of candle wax with oxygen, and that fire is a necessary precondition for illumination.

Beliefs concerning the propagation of light were revealed in discussion about a lighted candle placed on the floor at the center of a room. Some students claimed that the light intensity dropped slowly until at a certain distance it vanished. Students who marked a line to show the limit beyond which light did not pass explained that the

candle could be seen from beyond this line since the flame was brighter than its surroundings. Others claimed that light was latent in the air, using this to explain the phenomenon of light after sunset. A third belief was that light intensity falls as a function of colour, so that blue light doesn't travel as far as yellow light. The rate of change of intensity was also seen to depend on the velocity of light, which in turn depends on the power of the source. This was used to explain why a stronger source is visible from a greater distance.

Light was seen as composed of particles, of waves, as a combination of rays and a sea of light, and as being associated with heat.

Sight, it was believed, depended upon the reflection of light from bodies or upon light filling the surrounding space. Few students related sight to light entering the eye.

Light is believed to have a particular colour either because its source, such as a heated filament, is coloured, or as a result of passing through coloured glass. This may explain a student prediction that a blue object illuminated by red light would appear violet, rather than black, just as a mixture of blue and red paint appears violet.

At this stage of the study, having categorized some student beliefs about light, we identified three basic beliefs used in explaining light phenomena.

The first was the notion of a hot body radiating a flowing material or stream of tiny particles which can fill, and remain in, the air (light conservation as in the case of twilight). Light colours were seen to mix in the same way as do the colours of liquids. (22% of students).

The second was the notion of a hot body emitting variously sized particles, each size having a specific colour. 'White' particles easily change to particles of another colour, but the reverse process is more difficult since other colours are darker, and therefore more dominant, than white. In this view the colour of an object seen is a simple mixture of the colours of the object and of the illuminating light. A lab experience in which students saw that a red object viewed through a blue filter seemed black, did nothing to change this view of colour. (18% of students).

The third notion was that particles emitted by a chemical reaction are able to fill space but that their movement is retarded by the air, by some sort of friction. This would explain the belief that light from a source such as a candle flame propagates only to a limited distance. (22% of students).

These basic approaches to light all explain observed phenomena in materialistic terms in which the behaviour of light is reminiscent of the behaviour of a gas. This is the case not only for the 62% of students whose basic assumptions form the three clusters mentioned above, but for most of the other students too. We believe that students did not arrive at such a view of light as an outcome of formal learning, but that this view exemplifies a naive materialistic way of thinking about natural phenomena.

A NAIVE STANDPOINT AND FORMAL LEARNING

A series of laboratory learning situations was developed in which experimental phenomena examined by the student were accompanied by real time computer based analysis. The basic equipment included: an IBM PC; an A<->D converter; light and temperature sensors; a step motor controlled from the keyboard to move the sensors along a bench or to rotate them; a set of light sources; and a 3cm micro wave kit.

The lab-pc interface simultaneously provides empirical and analytical information. This two fold presentation increases what Posner et al. (1982) call the intelligibility of the physical concepts involved. On the basis of the simultaneous juxtaposition of both kinds of information the student is able to build, modify and rebuild explanatory hypotheses. In this resides the power of the system as an agent for change of students' conceptual frameworks since it provides for the immediate testing of ideas against experience. Unlike analysis in the traditional school laboratory experiment, here analysis becomes an integral part of the experiment.

The experiments covered: (1) the dependence of light intensity on the source, on distance, and on direction; (2) the notion of field; (3) the dependence of intensity on type, colour and thickness of an absorbing medium; (4) the relationship between absorbtion and temperature of a light transmitting medium; (5) polarization; (6) diffraction; (7)interference; and (8) the behaviour of 3cm electro magnetic waves.

Each experiment was planned to force students to draw inferences about relations among variables, and the process of learning from an experiment was seen as the process of learning to draw the inferences. The kinds of knowledge introduced by this teaching method were analysed and knowledge related to the experimental variables was identified.

We call knowledge on relationships among variables "relational knowledge" while "explanatory knowledge" refers to explanations as to why a particular relation holds rather than another. The first kind of knowledge was explicitly introduced in the experiments, while the students were encouraged to develop the second. For instance, in one experiment in which students examined the intensity of illumination from a light source, the relational knowledge found was $I= K/R2$. The related explanatory knowledge which students were expected to develop is that light propagates spherically. Both kinds of knowledge are significant, but our interest is mainly in the second.

Relational knowledge was examined by questionnaire, explanatory knowledge was examined by interviews during the experiments. Questions were designed to explore how students made sense of their experimental outcomes and how they defended their explanations.

In one experiment, for example, a light source heated two identical glass beakers, one containing strong black coffee, and the other weak coffee, during the same time period. The rise in temperature of the black coffee was found to be greater. Students explained that heat was created by friction between light particles and coffee particles and that since the black coffee was denser, friction was greater. One

student argued for this explanation by saying that "You can't see the light so clearly in fog because it is much slower than usual". When asked about the heat that should be produced, he said that the fog is so much larger than the cup of coffee that you don't feel it.

UNDERLYING EXPLANATORY BELIEFS

We found that students' explanations of experimental results were governed, in the main, by the following ideas:

The light sensor differs from the eye in that it "sees" only if the light is directed toward the sensor. This indicates an assumpion that light does not have to enter the eye in order for us to see.

Light is absorbed by material and fills it. Just as gas fills a container, so light enters and takes the container's shape. A material is bright if the particles of light stay inside it.

Particles of light can heat a medium through which it passes as a result of friction with particles of the medium..

The naive material particle conception of light seems to govern the learning of new concepts. Though relational knowledge was learned very well, this was not the case for explanatory knowledge. We believe that the tenacity with which students hold on to non science beliefs despite formal learning of science, results at least in part from the stress teachers place upon relational rather than explanatory knowledge.

In continuing this study, we now plan to construct a taxonomy of materialistic beliefs commonly used in the development of explanatory knowledge.

BIBLIOGRAPHY

Disessa, A.: Phenomenology and the evaluation of intuition: Working paper, M.I.T. division for study and research in education, 1981.

Driver, R., Bell, B.: Student thinking and the learning of science: A constructivist view. In: School Science Review, 1986, 443-456.

Lovell,K.: The relevance of cognitive psychology to science and mathematics education. In: Archenhold et al. (Ed.): Cognitive development research in science and mathematics education. Proceedings of an international seminar. University of Leeds, 1980.

Pope, M., Gilbert, J.: Personal experience and the construction of knowledge in science. In: Science Education 67 (2), 1983, 193-203.

Posner, G.J., Strike, K.A., Hewson, P.W., Gertzog, W.A.: Accomodation of a scientific conception: Toward a theory of conceptual change. In: Science Education 66, 1982, 211-227.

Resnick, L.B., Chi, M.T.H.: *The psychology of science learning: A post-Piagetian interpretation. Paper prepared for the NSTA yearbook on informal science education.*

Signalling Importance in Spoken Narratives: The Cataphoric Use of the Indefinite This

Morton A. Gernsbacher
Department of Psychology
University of Oregon

Suzanne Shroyer
Department of Linguistics
University of Oregon

In every narrative, many nouns are introduced. These can be participants, settings, props, and so forth. Once introduced, some of these people, places, or things play a pivotal role in the subsequent narrative; others, once introduced, are never to be mentioned again. Those people, places, or things that are mentioned more frequently we consider more topical (Givon, 1983), foregrounded (Chafe, 1974), or focused (Grosz, 1981). Those that are mentioned less often we think of as being supporting characters, backgrounded, or less focused.

While appreciating these distinctions, we assume that they map onto one psychological dimension: importance. Presumably concepts that are topical, foregrounded, or focused are perceived (by both the speaker and the listener) as being more important than concepts that are more peripheral, backgrounded, or less focused.

We also assume that discourse situations, like the telling of oral narratives, involve a functional exchange between speakers and listeners. Speakers use certain devices -- such as topicalization, foregrounding, and focus -- to convey certain properties -- such as importance. Presumably, on the receiving end, listeners are sensitive to these devices.

In this paper we explore one such device. It is the use of the unstressed, indefinite article this to introduce new noun phrases. We suggest that speakers use the indefinite this to signal their listeners that specific nouns are going to play a pivotal role in the upcoming discourse.

Furthermore, we also suggest that the indefinite this operates as a cataphoric device to increase the concepts' accessibility in the listeners' mental representations. We envision cataphoric devices as complements to anaphoric devices; whereas anaphoric devices (e.g., pronouns, repeated noun phrases, and the like) enable access to previously mentioned concepts, we suggest that cataphoric devices improve access to subsequently mentioned concepts. We will have more to say about how these devices might work later. But before we do, we shall further describe the cataphoric device that we studied in the present research.

The Indefinite This

Most of us are familiar with the indefinite article this; for example, it's typically used to introduce concepts in jokes, as in, "So this man walks into a bar" or "So a man walks into a bar with this parrot on his shoulder."

The indefinite this is interesting for a couple of reasons. First, it's a relative newcomer to English; Wald (1983) suggests that its use dates back only to the late 1930s. Second, the indefinite this occurs considerably more frequently in informal, spoken, dialects than formal or written ones, although many prescriptive grammarians dictate that it's unacceptable in any dialect (Prince, 1981).

The indefinite this shouldn't be confused with the deictic this ("Look at this") or stressed this ("I want this one"). In contrast to the unstressed, indefinite this, the deictic and stressed this are definite (Perlman, 1969). According to Prince (1981), the classic test of indefiniteness is occurrence in the existential-there construction. As (1) through (3) demonstrate below, the indefinite article this and the indefinite article a/an pass this test, whereas the definite article the fails.

(1) There was this guy in my class last quarter.
(2) There was a guy in my class last quarter.
(3) *There was the guy in my class last quarter.

By definition, the indefinite this -- like the indefinite a/an -- is used to introduce new concepts into a discourse.[1] Of the 243 occurrences of the indefinite this that Prince (1981) observed in Terkel's (1974) book, Working, 242 introduced a new concept, and the only exception was arguably introducing the same lexical form but with a different referent.

But more interestingly, in 209 of the 242 occurrences that Prince (1981) observed, the noun introduced with the indefinite this was referred to again, and as Prince said, "within the next few clauses". Wright and Givon (1987) quantified this observation more explicitly. They recorded eight- and ten-year olds telling one another informal stories and jokes. Wright and Givon found that when the children introduced nouns with the indefinite this, they referred to those nouns an average of 5.32 times in the subsequent 10 clauses that they produced; in contrast, when the children introduced nouns with the indefinite a, they referred to those nouns an average of only .68 times in the next 10 clauses. These data suggest that speakers use the indefinite this to introduce new concepts that are going to play a pivotal role in the subsequent narrative. Thus, the indefinite this serves as a cataphoric signal of importance.

Research Question

Our question in this research was whether listeners are sensitive to this cataphoric device. That is, does introducing a concept with the indefinite this, as opposed to the more typical indefinite a, increase that concepts' accessibility in the listener's mental representation?

To answer this question, we conducted the following experiment. We auditorily presented several informal narratives to native English speaking subjects. We told our subjects that at some point in each narrative the narrator would stop talking; when this happened, it was the subjects' job to continue telling the narrative. We constructed our narratives so that the last clause of each introduced a new noun phrase. We will refer to these nouns as "critical" nouns. We manipulated whether each critical noun was introduced by the indefinite this or the indefinite a. Below is an example of one of the narratives we constructed:

> I went to the coast last weekend with Sally. We'd checked the tide schedule 'n we'd planned to arrive at low-tide - 'cuz I just love beachcombin'. Right off, I found 3 whole sandollars. So then I started lookin' for agates, but I couldn't find any. Sally was pretty busy too. She found this/an egg ...

We proposed that accessibility would be manifested in our subjects' continuations in three ways: Frequency of Mention, Immediacy of Mention, and Referential Markedness. By Frequency of Mention we simply meant how frequently the subjects referred to the critical nouns. Presumably, the more accessible the concept was in the subjects' mental representations, the more frequently they would talk about it.

By Immediacy of Mention we meant whether the subjects would mention the critical noun in the first clauses that they produced. Presumably, the more accessible the concept was in the subjects' mental representations, the more likely they would be to mention it immediately (Gernsbacher & Hargreaves, 1988).

By Referential Markedness we meant how likely it was that subjects would refer to the critical nouns with more marked forms of reference, such as noun phrases, versus less marked forms, such as anaphoric pronouns. Several psycholinguists and linguists have shown that speakers' preferences for referential markedness is inversely related to their intended focus or topicality (Fletcher, 1984; Givon, 1985, Marslen-Wilson, Levy, & Tyler, 1982; Sidner, 1983). That is, speakers use less marked forms (e.g., zero anaphora and pronouns) for more focused or topical concepts. Similarly, we proposed that our subjects would use less marked forms for more accessible concepts. In other words, the more accessible the concept, the more likely the subjects would refer to it with a less marked form of reference. Our methodology and results are described more fully below.

Experimental Methods

Our subjects were 45 undergraduates at the University of Oregon who participated as one means of fulfilling a course requirement. All were native American English speakers.

Our materials were 20 experimental and 4 filler narratives. They ranged in length from 57 to 153 words with an average length of 93.7 words. All the narratives were written in a very informal, conversational dialect. The 20 experimental narratives were randomly ordered and intermixed with the 4 filler narratives. The purpose of the filler narratives was to disguise the occurrence of so many <u>this</u>-introduced noun phrases in the final clauses of the experimental narratives.

All 24 narratives were recorded by a college-aged male who we kept naive to the experimental hypotheses. Our narrator recorded two tapes: On one tape, half the experimental narratives were recorded in their <u>this</u>-introduced form, while the other half were recorded in their <u>a</u>-introduced form. On the other tape, the reverse was true. One indication of our narrator's naivete was that he inadvertantly recorded two of the experimental narratives in their <u>this</u> form on both tapes. Although we didn't realize this mistake until after collecting the data, we dropped these two narratives from our analyses.

During the experiment, each subject sat in a sound proof room. The subject listened to our recorded narratives, as well as our instructions, over a set of headphones. The subjects were told that they would hear the beginnings of twenty-four stories and that they should complete them as they felt the narrator of the stories would have done. At the end of each narrative, the subjects were given 20 seconds to tell their continuation. When the 20 seconds were up, the subjects heard a single tone. They were to stop talking then if that was convenient, but if it wasn't, they could continue for an additional ten seconds, at which time they heard two tones. After the two tones, they were then given a

fifteen second break before the next narrative began. To get accustomed to the experimental task, the subjects first practiced on two narratives.

Experimental Results

Each subject's continuations of the experimental narratives were transcribed according to the methods of Ochs (1979). In these transcriptions, clause boundaries were marked on the basis of finite verbs and intonation groups (Chafe, 1980). Two judges scored the transcripts blind to both the subjects' identities and the narratives' experimental condition, (i.e., whether the critical nouns were introduced with this versus a).

Frequency of Mention. We first measured how frequently subjects mentioned the critical nouns. When the critical nouns were introduced with the indefinite this, subjects mentioned the nouns in 22% of the clauses that they produced; in contrast, when the critical nouns were introduced with the indefinite a, subjects mentioned the nouns in only 16% of the clauses they produced. This difference was statistically reliable, $t(1,17) = 2.312$, $p < .03$.[2] In fact, in 17 of the 18 narratives we observed the expected difference (i.e., subjects more frequently mentioned the critical nouns when they were introduced with this versus a).

As an illustration, below is Subject #03's continuation for the same narrative that we presented as an example above. This subject heard the narrative with the critical noun egg introduced with the indefinite this. As illustrated in his continuation, this subject frequently refers to the critical noun.

> 'N it looked like it came from a lizard or something or maybe a turtle, but we couldn't tell if it had hatched or not so we put it back where we found it just in case it was still alive

In contrast, below is Subject #30's continuation for the same narrative, but Subject 30 heard the narrative with the critical noun egg introduced with the indefinite a.

> But what I really wanted to find was a whole crab shell. Y'know you can hardly ever find those. You always find just bits and pieces. It's like someone deliberately comes up 'n crunches 'em all before 'ya get there or somethin'. But, beachcomin's a favorite thing, even if I don't find any crab shells.

Immediacy of Mention. The second property that we measured from our subjects' continuations was their likelihood of mentioning the critical noun in the first clause that they produced. When the critical nouns were introduced with the indefinite this, subjects mentioned those nouns in 46% of the first clauses of their continuations; in contrast, when the critical nouns were introduced with the indefinite a, subjects mentioned those nouns in only 37% of their first clauses. This difference was also statistically reliable, $t(1,17) = 2.03$, $p < .055$. In fact, we observed this expected difference in 15 of the 18 experimental narratives.

Referential Markedness. Our third measure was the subjects' choice of referent for the critical nouns. To quantify this effect, we attended only to those continuations where subjects did in fact refer to the critical nouns. And in those continuations, we attended only to the subjects' first reference to the critical nouns. There were a total of 533 of these first references; 376 were pronouns (the less marked form of reference)

and 177 were noun phrases (the more marked form of reference). Of the 376 pronouns, 57% were references to this-introduced critical nouns, and the remaining 43% were references to a-introduced critical nouns. Thus, subjects were more likely to select the less marked form of reference to refer to this-introduced critical nouns. The opposite pattern emerged with the noun phrases (the more marked form of reference). Of the 177 noun phrases, 54% were references to critical nouns that had been introduced with a, and the remaining 46% were references to critical nouns that had been introduced with this. Thus, subjects were more likely to select the more marked form of reference to refer to a-introduced critical nouns. This association was statistically different from what would be expected by chance, $X^2(1) = 6.014$, $p < .013$.

Finally, we also encountered a rather interesting datum. A small proportion of the time (2.83%), subjects committed what we will refer to as a "switch." In the narrative they heard, the critical noun was introduced with the indefinite this, but when they first mentioned the noun, they switched the this to an a. Or conversely, they heard the critical noun introduced with the indefinite a, but they switched the a to a this. We found that switching this to a was very rare. That situtation occurred on only 13% of the switches (or .388% of the continuations). In contrast, swtiching a to this occurred significantly more often; that situation accounted for 87% of all the switches (or 2.44% of the continuations), $t(1,17) = 3.578$, $p < .002$. What this pattern suggested to us was that subjects felt that they had to signal importance with the indefinite this before they could focus, foreground, topicalize or otherwise elaborate on the nouns in their own continuations.

Conclusions

To summarize our results, we found that our introducing noun phrases by the indefinite this versus the indefinite a greatly affected our subjects' continuations: When the critical nouns were introduced with this, subjects mentioned the nouns more frequently, often within the first clauses that they produced, and typically via pronouns. In contrast, when the critical nouns were introduced with a, subjects mentioned the nouns less frequently, and typically via full noun phrases.

These data suggest that listeners are indeed sensitive to the rapidly developing use of the indefinite this to signal potentially important information. Concepts introduced by a speaker with the indefinite this are more accessible to the listener. In this way, the indefinite this appears to be operating as what we are calling a cataphoric device.

Indeed, Prince (1981) has suggested that the indefinite this parallels a device in American Sign Language in which a signer establishes an absent third person on his or her right so that the signer might later refer to that individual; an absent third person who is not intended to be subsequently referred to is not established this way. Clearly, this ASL device is also operating cataphorically.

Another device that might work cataphorically in spoken English is intonation (or spoken stress). Cutler (1976) found that initial phonemes were recognized faster when they began words that carried their sentences' stress (see also Shields, McHugh, & Martin, 1974). Cutler (1976) also found that it wasn't just the improved acoustic quality that effected this advantage: She found the same results when she exchanged (by tape splicing) the stressed word with the same word from an unstressed environment. In other

words, it was the intonation contour that directed the listeners toward the stressed items.

Similarly, Bock and Mazzella (1983) found that stressing the word Arnold in a sentence such as,

(4) Arnold didn't fix the radio.

as opposed to the word fix, made it easier for listeners to understand a subsequent sentence like

(5) Doris fixed the radio.

So both Cutler's (1976) and Bock and Mazzella's (1983) studies suggest that intonation signals information focus, or what we have been calling importance. Thus, intonation seems like a good candidate for a cataphoric device.

How do cataphoric devices work? That is, what are the mental processes underlying our suggestion that cataphoric devices increase their concepts' accessibility in listeners' mental represenations? One hypothesis is based on some of our recent findings concerning how anaphoric devices work (Gernsbacher, 1988). We have found that anaphoric devices (e.g., pronouns, repeated noun phrases, and the like) enable access to previously mentioned concepts via two mechanisms: Enhancement, which increases the activation level of the rementioned concept's mental representation, and Suppression, which decreases or dampens the activation of other concepts' representations. The net effect of both mechanisms is that the anaphorically mentioned concept is activated at a different level than other concepts.

Similarly, we suggest that cataphoric devices may use either or both of these two mechanisms. That is, a cataphoric device may improve its concept's accessibility by enhancing the mental representation of that concept. Or a cataphoric device may improve its concept's accessibility by suppressing the activation of other concepts. We are excited about empirically exploring these possibilities.

Footnotes

[1]A few culturally shared or contextually unique concepts are introduced with a definite article, for example the sun, the President, or "I walked into the house and the stereo was blaring" (Chafe, 1987; Clark & Marshall, 1981, Givon, 1986).

[2]Our statistical analyses treated the 18 experimental narratives, or more technically the 18 critical nouns introduced in the 18 experimental narratives, as our units of analysis. This approach is more conservative than the typical subject-based analyses (Clark, 1973).

References

Bock, J.K., & Mazella, J.R. (1983). Intonational marking of given and new information: Some consequences for comprehension. Memory & Cognition, 11, 64-76.

Chafe, W. (1974). Language and consciousness. Language, 50, 111-133.

Chafe, W. (1980). (Ed). The pear stories: Cognitive cultural, and linguistic aspects of narrative production. Norwood, NJ: Ablex.

Clark, H.H. (1973). The language as a fixed-effect-falacy: A critique of language statistics in psychological research. Journal of Verbal Learning and Verbal Behavior, 12, 335-359.

Clark, H.H., & Marshall, C.R. (1981). Definite reference and mutual knowledge. In A. Joshi, B. Webber, & I. Sag (Eds.), Elements of discourse understanding (pp. 10-63). Cambridge: Cambridge University Press.

Cutler, A. (1976). Phoneme-monitoring reaction time as a function of preceding intonation contour. Perception & Psychophysics, 20, 55-60.

Fletcher, C. R. (1984). Markedness and topic continuity in discourse processing. Journal of Verbal Learning and Verbal Behavior, 23, 487-493.

Gernsbacher, M.A., & Hargreaves, D. (1988). Accessing sentence participants: The advantage of first mention. Journal of Memory and Language, in press.

Givon, T. (Ed.) (1983). Topic continuity in discourse: A quantitative cross-language study. Amsterdam: J. Benjamins.

Givon, T. (1985). The pragmatics of referentiality. In D. Schiffrin (Ed.), Meaning, form, and use in context. Washington, DC: Georgetown University Press.

Grosz, B. (1981). Focusing and description in natural language dialogs. In A. Joshi, B. Webber, & I. Sag (Eds.), Elements of discourse understanding (pp. 84-105). Cambridge: Cambridge University Press.

Marslen-Wilson, W., Levy, E., & Tyler, L. K. (1982). Producing interpretable discourse: The establishment and maintenance of reference. In R.J. Jarvella & W. Klein (Eds.), Speech, place and action (pp. 339-378). Chichester: Wiley.

Perlman, A. (1969). 'This' as a third article in American English. American Speech, 44, 76-80.

Prince, E. (1981). On the inferencing of indefinite-this NPs. In A. Joshi, B. Webber, & I. Sag (Eds.), Elements of discourse understanding (pp. 231-250). Cambridge: Cambridge University Press.

Shields, J.L., McHugh, A., & Martin, G. (1974). Reaction time to phoneme targets as a function of rhythmic cues in continuous speech. Journal of Experimental Psychology, 102, 250-255.

Wald, B. (1983). Referents and topic within and across discourse units: Observations from current vernacular English. In S. Kelien-Andreu (Ed.), Discourse perspectives on syntax (pp. 91-116). New York: Academic Press.

Wright, S., & Givon, T. (1987). The pragmatics of indefinite reference: Quantified text-based studies. Studies in Language, 11, 1-33.

Acknowledgments. This work was supported by National Science Foundation grant BNS 85-10096. We thank Carrie Clarke and Kathy Varner for their help in transcribing the subjects' continuations, and Ellen Galloway for her help in testing the subjects. Please address correspondence to Dr. M.A. Gernsbacher, Department of Psychology, University of Oregon, Eugene, OR 97403-1227.

PLANNING AND IMPLEMENTATION ERRORS IN ALGORITHM DESIGN

Wayne D. Gray,
U. S. Army Research Institute

Albert T. Corbett, & Kurt Van Lehn
Carnegie-Mellon University

Introduction

This study examines the algorithm design process for 59 LISP programmers who tackle a classic artificial intelligence search problem for the first time. Programmers were asked to code a single function called *descendent*, that was of average length and that performs a depth-first search over an hierarchy. This was a fairly difficult task. In this paper, we outline a set of basic planning steps for designing this algorithm and examine variations in the 59 solutions that reflect divergences at different steps.

Various aspects of algorithm and software design have been studied before and this study owes much to those efforts (for example, 1, 3, 8, 10, and 14). Especially relevant are those studies that have emphasized "bugs" (9, 12, 13).

The current study looks for evidence of **plan** or **implementation** failures (bugs) at each step in the design process. Plan failures are indicated by the use of plans inappropriate to the current problem (5) and are a type of negative transfer (7). Implementation failures result from the failure to correctly translate a plan into the programming language (11).

The Study.

The Programmers.

Fifty-nine students were drawn from four LISP courses and one course on cognitive science. All students used the same introductory LISP textbook (2) and completed at least those lessons in the LISP Tutor (4) that covered the functions and basic control structures required in this study. At the time of the study, no student had attended lectures or read the textbook chapter on search techniques.

The Problem Specification

Programmers were asked to write a depth-first search[1] function that took two arguments and determined whether the second argument was a descendent of the first. If so, the function returned "**t**," otherwise "**nil**." The problem statement was accompanied by the hierarchy in Figure 1. Given this example, ***(descendent 'Bill 'Frank)*** should return ***t***, while ***(descendent 'Bill 'Joe)*** should return ***nil***.

Two important constraints were imposed upon the task. First, programmers were not given any information about how hierarchies were represented in LISP. Their only means for searching hierarchies was an expansion function (called ***expand***) that accepted one node and returned a list of the node's immediate descendents. For example, from Figure 1 ***(expand 'Bill)*** would return ***(Julia Mike)***. Second, programmers were to write an iterative function with no recursive function calls. (More specifically, they were required to use a **let/loop** construction rather than **do**, to further standardize the goal state).

Finally, the problem description recommended that a local variable be used to save the list of nodes that was generated by the expansion function until they could be checked. The following function definition satisfies these constraints:

---

[1] A depth-first search of a tree moves down whenever possible to get the next node, and only moves back up and over when it is not possible to move down. A depth-first search of the tree in Figure 1 might check the nodes in the following order: Harry, Jane, Joe, Diane, Bill, Julia, Frank, Anne, Susan, Mike.

```
(defun descendent (ancestor target)
  (let ((queue (list ancestor)))
    (loop
      (cond ((null queue) (return nil))
            ((equal (car queue) target) (return t)))
      (setq queue (append (expand (car queue)) (cdr queue))))))
```

In this function, ***queue*** holds a list of nodes that have been accessed and need to be checked. Each time through the loop ***queue*** is tested. If it is empty, the search has failed and the function returns ***nil***. If it is not empty, the first node in ***queue*** is compared to the target and if they match the function returns ***t***. Otherwise, the first node is removed from ***queue*** while its immediate descendents are added to the list. In the case of Figure 1, if ***queue*** held the list ***(Jane Bill)*** and the first node ***Jane*** was tested and rejected, then on the next cycle ***queue*** would hold ***(Joe Diane Bill)***.

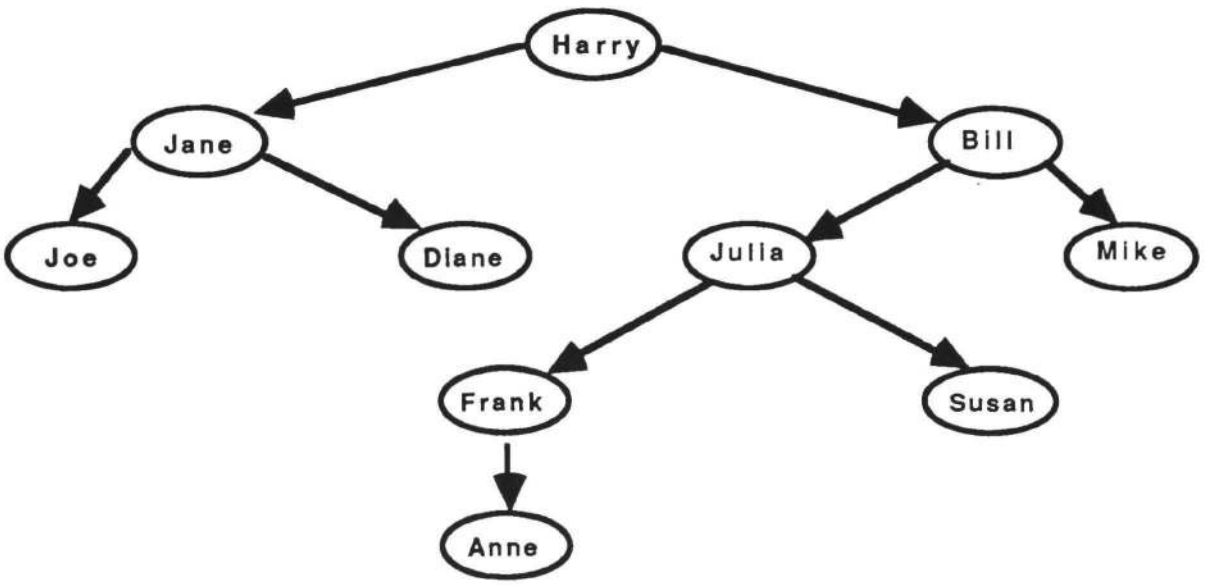


Figure 1
Example Hierarchy given to programmers coding "descendent."

Algorithm Design

At first glance, the solution consists of a simple two step cycle:[2] 1. ***Get the Next Node***, 2. ***Test the Node***. The second step is easy to implement, but the ***get the next node*** step is quite difficult. If programmers were given a function that takes a node and simply returns the appropriate next node to test, then this problem becomes trivial. Instead, the function, ***expand***, accepts one node and returns a ***list*** of all the descendents of that node.

expand imposes a constraint on the solution that is not an intrinsic part of a depth-first search over a tree. The function returns a list on each cycle rather than directly accessing individual nodes in the tree. This expansion function constraint leads most directly to an algorithm that is essentially recursive. If we label the algorithm ***CHECK-LIST***, we can represent its recursive structure as the sequence of operations shown in Figure 2.

[2]There are, in fact, additional issues the student must address before completing the design. For example, each cycle must also contain a test to see if the network is exhausted.

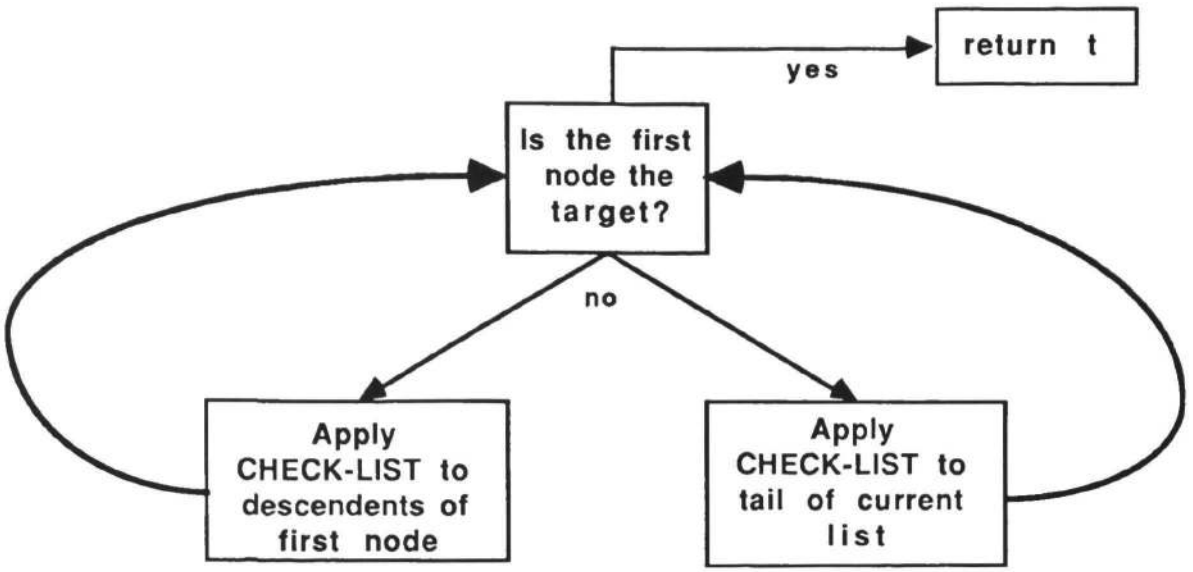


Figure 2
Simplified Recursive Solution

The corresponding LISP function might be coded as follows:

```
(defun descendent (given target)
    (check-list (list given) target))

(defun check-list (current-list target)
    (cond ((null current-list) nil)
          ((equal (car current-list) target) t)
          (t (or (check-list (expand (car current-list)) target)
                 (check-list (cdr current-list) target)))))
```

This solution conforms to a type of recursion, ***car-cdr*** recursion, with which the programmers were familiar. The iterative constraint blocked this solution, of course, and sets up the most demanding aspect of the planning process.

Students are required to discover an iterative solution which is isomorphic to this recursive concept. That isomorphic iterative algorithm can be specified as shown in Figure 3. This specification gives rise to the definition of descendent presented earlier.

While the results of the iterative solution are isomorphic to the recursive solution, there is an important conceptual difference. The recursive solution does not require building a new list structure, the iterative solution does. In Figure 2, the recursive function, **CHECK-LIST**, is applied to both the existing list structure and the list structure returned by ***expand***. In contrast, the iterative solution requires building a new list on each iteration. As shown in Figure 3, on each cycle, two different operations are performed upon the list and the results of these two operations are combined into a new list. (For this reason the iterative solution will be referred to as the ***list-building*** algorithm.)

This characterization of the iterative solution suggests three steps in the design process that may cause difficulties and lead to bugs in the students' code: (1) the transition from thinking in terms of individual nodes to thinking in terms of expansion lists, (2) the recognition that a depth-first search requires a solution analogous to ***car-cdr*** recursion, and (3) the recognition that, unlike the recursive solution, the iterative algorithm requires that a new list be built on each cycle.

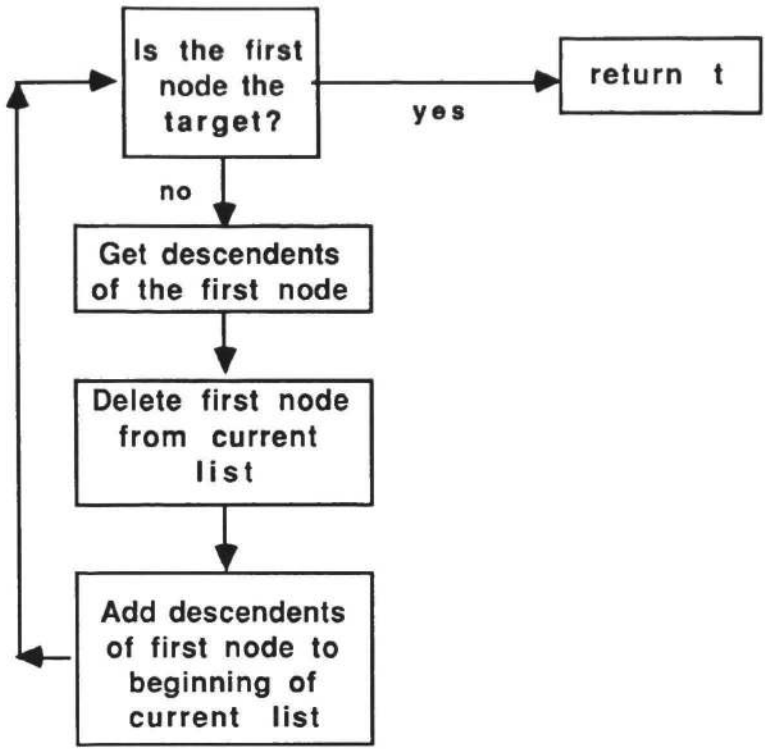


Figure 3
Simplified Iterative Solution

Simulating the algorithm.

A subset of 23 programmers were asked to perform a paper and pencil simulation of the function prior to coding it. They were asked to simulate the function call ***(descendent 'Harry 'Frank)***, by writing down the initial value of the variable ***queue*** and then, for each iterative cycle, writing the node that would be checked, the immediate descendents of the node, and the new value of ***queue***. Thus, for one cycle, the to-be-checked node is ***Jane***, its immediate descendents are ***(Joe Diane)*** and the new value of ***queue*** is ***(Joe Diane Bill)***. Programmers were given feedback to ensure that they simulated the function correctly. Programmers who were guided directly through the iterative solution would not be expected to conform to the standard algorithm design sequence proposed in the earlier section.

Procedure

Programmers were asked to *talk-aloud* while coding the function, that is, to report what they were thinking as they worked (6). At the beginning of the session, each programmer was given practice in talking aloud and then read an abridged version of the search chapter from their LISP textbook that described hierarchies, depth-first search, and expansion functions. This abridged version did not discuss how to implement a search function.

After reading the text, the programmer was given the problem description and asked to write down the order in which the nodes should be checked, to ensure that s/he understood the concept of a depth-first search. Then the programmer simulated the function if s/he were one of the 23 programmers in the simulation condition. Finally, the programmer coded the function on a computer terminal while talking aloud. The programmers worked on the function until either (1) they were satisfied with their solution or (2) gave up on it or (3) one hour elapsed. Programmers were not able to test their function in the course of coding it.

Results and Discussion.

This was a relatively difficult task. Of the 59 programmers, only 3 wrote functions that would work with no modifications. Thirty-five solutions contained only minor implementation[3] and/or planning errors, 17 contained major planning errors, but only 4 were completely uninterpretable. Viewed to emphasize the positive, 55 of the 59 programmers wrote functions that contained an interpretable control structure that in principle would have searched some or all of the tree. The interesting question, however, is not how many programmers coded the function correctly, but what the final code reveals about difficulties in planning and implementation.

Table 1
Comparison of Algorithm Use by Type of Training

	simulation	control	totals
list-builders	19	18	37
other	1	15	16
totals	20	33	53

Fifty-three of the 59 solutions were rated as trying to solve the correct problem.[4] Of these 53, 38 apparently worked through the planning process successfully. (Of these, 37 programmers generated solutions that are consistent with the *list-building* solution described above and one designed a unique solution consistent with the problem statement.) The simulation manipulation reveals whether the basic difficulty is in generating the plan or implementing it. Of the 20 programmers who simulated an example of the list-building algorithm, 19 (95%) generated code that conformed to the algorithm. Only 18 of 33 (55%) programmers in the control condition conformed to the list-building algorithm. The results are shown in Table 1. A chi-sq test showed these differences to be significant (Chi-sq [1, **n**=53] = 7.85) ($p < .05$), indicating that the simulation trained programmers used the list-building algorithm more often then would be expected and suggesting that when a solution deviated from the algorithm it was largely because of difficulty in generating the plan, rather than an implementation failure.

Planning Steps: Node Testing.

As described above, the node test is easy to plan and implement and all but one solution contained a test that compared the nodes in the tree to the second argument. However, there is an interesting difference among the 53 programmers who wrote interpretable code. Rather than testing a single node on each cycle, (using ***equal***), 14 of the programmers compared the target to a whole list of nodes (using ***member***).

---

[3]Implementation errors include both syntactic and semantic errors. Examples of minor syntactic errors include misplace or missing parentheses, or inappropriate use of quotes. Examples of minor semantic errors include substituting a similar, but inappropriate LISP function for the correct one (such as using ***cons*** for ***append*** in the update) or initializing the local variable to a node (a LISP atom) when it should have been initialized to a list containing the node.

[4]We could make no sense of 4 solutions, so these were eliminated from further consideration. Likewise, we have not included two unique, but non-depth-first search algorithms. These exclusions leave 20 programmers with simulation training and 33 without for a total of 53.

We hypothesize that the choice ***member*** versus ***equal*** represents a planning, not an implementation, bug. Our programmers were very familiar with both ***member*** and ***equal***. At the implementation level it seems unlikely that one would be mistaken for the other. In contrast, the use of ***member*** may represent the transfer of a very natural perceptual strategy. If we were to physically retrieve a list of descendents (especially a short list), it would be nearly impossible not to scan the entire retrieved list and determine if the target is on the list. We hypothesize that the use of ***member*** is evidence that this *naive* plan has substituted for the node test plan required by the problem specification.

Planning Steps: Getting New Nodes.

The nature of the expansion function imposes constraints on the algorithm for getting new nodes. In particular, it imposes a list structure on the planning process and gives rise most naturally to a recursive solution. The ban on recursive function calls constrains the programmer to transform the recursive solution into an iterative solution that builds a list.

There is some evidence that a few programmers had difficulties with superimposing a list structure on the tree diagram. Specifically, 6 programmers generated solutions in which a local variable was processed in some contexts as if it stored a single node and in other contexts as if it stored a list. The remaining 47 programmers did not appear to have this difficulty.

Thirty-seven programmers employed the list-building iterative solution in generating new nodes. One programmer employed LISP *property lists* to generate a hierarchical structure that directly paralleled the diagrammatic tree structure and used these properties to structure the search process. This solution is fascinating since it diverges widely from the standard plan and hints at the actual size of the algorithm space. It will not be considered further, precisely because it does not cast light on the difficulties of the list-building plan. The final two categories represent fundamentally flawed variations of the list-building plan.

The first variation, coded by seven of the programmers, is a ***depth-first/dead-end*** search of the tree. In this algorithm, the first element in ***queue*** is searched and expanded in each cycle. However, the remainder of ***queue*** is discarded and ***queue*** is set equal to the expansion, as in the following LISP expression which would be substituted for the final line of the list-building solution:

```
(setq queue (expand (car queue))).
```

This solution searches down one branch of the tree (if the target is not found along the way) and then terminates. In the case of Figure 1, the nodes ***Harry***, ***Jane***, and ***Joe*** would be checked.

The second variation, coded by 7 different programmers, might be called a ***two-step*** algorithm. In this algorithm, ***queue*** is initialized to the expansion of the first argument. Then in every cycle the first element in ***queue*** is expanded, that expansion is tested (with a **member** test), and the element is removed from ***queue***. The following code, which would be substituted for the final two lines of the list-building solution, characterizes this algorithm:

```
(cond ((member target (expand (car queue))) (return t)))
(setq queue (cdr queue))
```

This solution will search the top two levels of the tree. At least some of the 7 programmers recognized that the solution was inadequate and tried to extend it with baroque yet futile additions, for example, by incorporating an inner loop to reach down another ply in the tree.

These categories are interesting in that the failure is closely linked to different aspects of the plan described earlier. The ***depth-first/dead-end*** solution may represent a failure to fully formulate the recursive plan. That is, this solution checks and expands the first node in the queue in each cycle, much as each call to a recursively defined function would, but completely fails to process the tail of the list, in effect omitting the ***cdr*** component of the ***car-cdr*** recursion.

The ***two-step*** approach, on the other hand, seems to be based on a fully specified recursive solution that is not correctly translated into an iterative solution. In this solution, the processing of

the tail of the list is structurally correct, as is the expansion of the *car*. However, the requirement to build a new list on each iteration is not recognized. This solution may directly reflect the programming experience of the programmers. All students learned how to code equivalent *tail recursive* and *list iteration* functions, and this component of the algorithm is coded correctly. On the other hand, while students also encountered *car-cdr* recursive functions, this experiment was their first experience in generating equivalent iterative functions.

Conclusion

This report is necessarily brief and by omitting discussion of various issues concerning both systematic and non-systematic deviations may not fully convey the degree of variability obtained across solutions in this experiment. Moreover, there remain response patterns that are difficult to evaluate simply on the basis of the final code. For example, some programmers test whether a node has descendents before adding the descendents to the queue although, given the definition of the function ***append,*** this step is unnecessary. It is unclear from examining the solutions whether this is a plan bug imported from naive notions of hierarchical search (see also 5), or an implementation bug tacked on because of uncertainty about how ***append*** works. The long term goal of this research is to develop a more detailed model of the algorithm design process, on the basis of keystroke data and tapes of the coding sessions.

Nevertheless, granting the wide degree of coding variability obtained in this study, it is possible to discern categories of errors that reflect not just implementation failures, but failures in predictable steps in the algorithm design process

References

1. Adelson, B., & Soloway, E. (1987). A model of software design. In M. Chi, R. Glaser, and M. Farr (Eds.), **The nature of expertise.** Hillsdale, NJ: Erlbaum.
2. Anderson, J. R., Corbett, A. T., & Reiser, B. J. (1987). **Essential LISP**. Reading, MA: Addison-Wesley Publishing Company, Inc.
3. Anderson, J. R., Farrell, R., & Sauers, R. (1984). Learning to program in LISP. **Cognitive Science, 8**, 87-129.
4. Anderson, J. R., & Reiser, B. J. (1985, April). The LISP tutor. **Byte**, pp.159-175.
5. Bonar, J., & Soloway, E. (1985). Pre-programming knowledge: A major source of misconceptions in novice programmers. **Human-Computer Interaction, 1.**
6. Ericsson, K. A., & Simon, H. A. (1985). **Protocol analysis: Verbal reports as data**. Cambridge, MA: MIT Press.
7. Gray, W. D., & Orasanu, J. (1987). Transfer of cognitive skills. In S. Cormier and J. Hagman (Eds.), **Transfer of learning**. Orlando, FL: Academic Press.
8. Jeffries, R., Turner, A. A., Polson, P. G., & Atwood, M. E. (1981). The processes involved in designing software. In J. R. Anderson (Ed.), **Cognitive skills and their acquisition.** Hillsdale, NJ: Erlbaum.
9. Johnson, W. L., & Soloway, E. (1984). Intention-based diagnosis of programming errors. **Proceedings of the 1984 Conference of the AAAI**, 162-168.
10. Kant, E. (1985). Understanding and automating algorithm design. **IEEE Transactions on Software Engineering, 11**, 1361-1374.
11. Moran, T. P. (1983). Getting into a system: External-internal task mapping analysis. In **Proceedings of the ACM SIGCHI Conference on Human Factors in Computer Systems**. Boston, MA.
12. Soloway, E. (1985). From problems to programs via plans: The content and structure of knowledge for introductory LISP programming. **Journal of Educational Computing Research, 1**, 157-172.
13. Soloway, E., Bonar, J., & Ehrlich, K. (1983). Cognitive strategies and looping constructs: An empirical study. **Communications of the ACM, 26**, 853-860.
14. Steier, D. M., & Kant, E. (1985). The roles of execution and analysis in algorithm design. **IEEE Transactions on Software Engineering, 11**, 1375-1386.

Conceptual Slippage and Analogy-Making: A Report on the Copycat Project

Douglas R. Hofstadter and Melanie Mitchell
Department of Psychology and Department of Computer Science
University of Michigan

In our research we are investigating the mechanisms underlying human analogy-making. We are developing a theory of these mechanisms, which centers on the interaction of perception with the associative, overlapping, and context-sensitive human conceptual system, and on how this interaction gives rise to "conceptual slippage" (the flexible translation of ideas from one framework to another), which is required for creative analogy-making. To test this theory, we are building a computer model called "Copycat", which is able to make analogies in a microdomain. Although the microdomain appears small and simple, it is surprisingly rich; extremely subtle analogies requiring great flexibility and creativity can be made in it, and we believe it is an excellent testbed for computer models of analogy-making. This paper describes the current state of our research, and shows in detail (using a series of screen printouts from two runs of the program) how Copycat's perceptual mechanisms interact with its conceptual system and allow it to describe situations and make analogies. Previous work on the Copycat project has been reported by Hofstadter (1984b, 1985), Hofstadter, Mitchell, & French (1987), Hofstadter & Mitchell (1988), and Mitchell (1988).

Copycat's microworld consists of the 26 letters of the alphabet and associated concepts; in it we construct analogy problems involving letter-strings. A simple problem is: If **abc** changes to **abd**, what is the analogous change to **pqrs**? Most people answer **pqrt**, using the rule "Replace the rightmost letter by its successor". However, if the target string were **ppqqrrss** rather than **pqrs**, that rule would yield **ppqqrrst**, which almost all people see as too rigid. The rule thus has to be "translated" to the new situation. But a different translation is needed for target **ssrrqqpp**, and still other translations for targets **mrrjjj** (in which numerical successorship plays the role of alphabetic successorship), **aababc** (extending the notion of successorship if the string is parsed **a-ab-abc**), **ace** (double successorship), and **xyz** (**Z** has no successor). A vast number of interesting problems can be constructed in this domain (see Hofstadter, Mitchell, & French (1987) and Hofstadter & Mitchell (1988) for collections of such problems, and see Hofstadter (1985) and Mitchell (1988) for discussions of how these problems relate to "real-world" analogy-making).

We will explain the workings of the current version of the Copycat program by presenting two series of annotated screen printouts from actual runs on two problems. First, we discuss some of the ideas behind the model. The first idea is nondeterminism, which permeates the workings of Copycat. The program accomplishes its goals by executing a very large number of small pieces of code, called "codelets", chosen probabilistically from a constantly changing pool. Thus not only does each run differ from every other run, but also many different answers can be reached for a given problem. Thus to show just one run for a given problem (as we have done) is somewhat misleading. We have chosen two fairly typical runs for the two problems, but readers should bear in mind that other answers are often produced, and many other routes to the shown answers exist.

Copycat's nondeterministic nature is based on the idea that analogy-making, like perception, is highly and asynchronously parallel. In perception and in Copycat, many processes take place concurrently. In Copycat, each process consists of many small codelets, and codelets of different processes are interleaved probabilistically. Each process has a dynamically evaluated importance, so that favored processes can run faster. This is accomplished by giving each codelet an "urgency" -- a number that determines its probability of being chosen from the pool of codelets waiting to run.

The second idea is the building-up of a coherent view. It is up to the program to build up an understanding of each letter-string on its own, and also of how strings are related. This is very similar to a perceptual process. At the outset, each codelet picked has a small region of a string as its focus, and it looks for any local structure of interest there. If so, it suggests that that structure

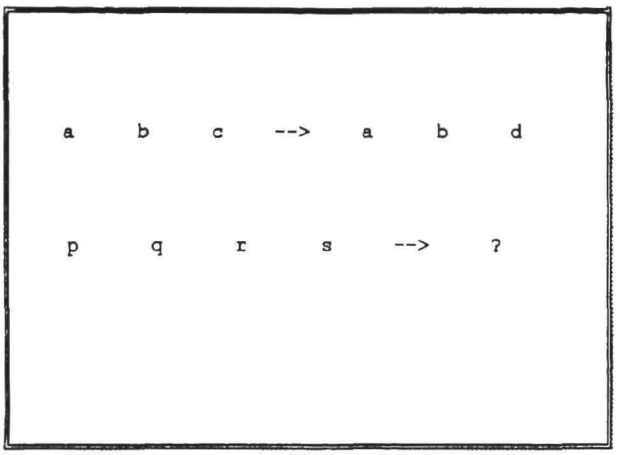
be officially recognized by another codelet. As many such codelets run, they gradually "annotate" individual letters and letter-strings, converting them from raw data into coherently understood structures. This process is very similar to the operation of the Hearsay II speech-understanding program (Erman et al., 1980), which took a raw speech waveform and allowed many processes to build higher-level hypotheses about it, upon which yet higher-level hypotheses could be built, at the top level of which emerged a totally semantic understanding of the utterance. The aim of Copycat is similar: to convert a raw letter-string into a totally understood situation.

Moreover, not only must Copycat knit together each separate string, it must also construct a coherent network of correspondences between the three given strings. These correspondences express Copycat's view of how certain parts of one structure map onto parts of another, without there necessarily being any one-to-one mapping involving all the parts. The build-up of local structure inside a given string tends to precede the build-up of correspondences between strings, but this is not an ironclad order; the nondeterminism allows these types of processes to take place concurrently. At the outset, intra-string processes are given higher urgencies, but as coherent views of individual strings gradually get built up, the urgencies of inter-string processes rise and so those processes become predominant. Thus activity gradually shifts from a local to a global scale.

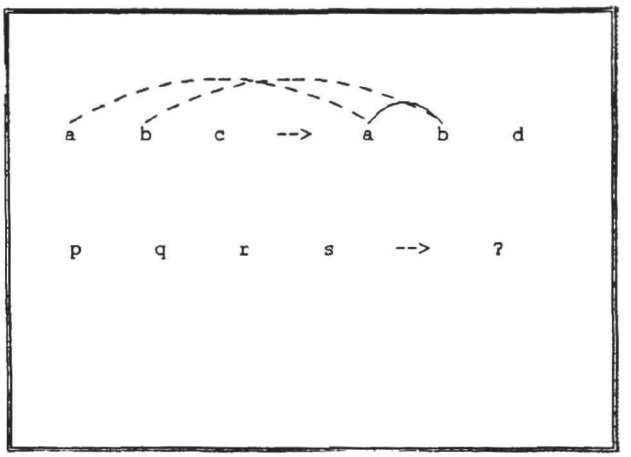
The third idea is that of conceptual distance and slippage. In any analogy worth the name, there are "conceptual slippages": mental correspondences made between things that are not identical. In Copycat, the plausibility of any such correspondence is determined by referring to a network of concepts called the "Slipnet", one of whose main functions is to define a "distance" between any chosen pair of concepts. The smaller the distance between the two concepts, the more plausible is a mapping in which they are considered counterparts (i.e., in which the one "slips" into the other). Of course, the smallest possible distance is zero -- when a concept is mapped onto itself. But an analogy in which all conceptual distances are zero would be a total identity. Thus non-trivial slippage is an essential ingredient of interesting analogy-making. As its name would imply, the Slipnet is the measure of all slippages.

The Slipnet is a dynamically changing network, in which conceptual distances change as a function of processing (thus as a function of context). There is a default setting of the Slipnet, in which concepts have "neutral" distances, but as certain slippages take place, they modify the distances between similarly-related concepts. For example, if a slippage between two concepts considered to be "opposites" is incorporated into an analogy, that shortens the distances between all pairs of "opposites" in the Slipnet, which tends to increase the likelihood of similar slippages. Each concept has a time-varying activation level: a function of the importance of the role the concept has been perceived as playing. When a concept is activated, its use in forming descriptions is encouraged. For example, if a group of any sort has been perceived, codelets attempting to create other groups of that sort will henceforth tend be more successful. Another crucial function of activation is that of determining the salience of all the objects in the situations at hand. Each object (letter, group of letters, entire string) has a number of descriptions, each of which consists of names of certain concepts in the Slipnet. To each description is attached a time-varying number that reflects how active those concepts are at the moment, and the object's salience is a simple function of those numbers for all its descriptions. Thus an initially unremarkable object can become strikingly salient if one or more of its descriptions involve highly activated concepts. The reason this matters is that codelets are highly biased towards acting upon salient objects. Thus there is an interesting reciprocal influence of the Slipnet and the perceptual processing of strings: the Slipnet determines what objects are most "interesting" as foci of processing, and the results of processing determine the level of activation of Slipnet concepts, which feeds back into the processing. Finally, activation spreads from a concept to neighboring concepts in the Slipnet, so that even if a concept is not directly involved in the situations, its conceptual closeness to concepts that are directly involved may cause it to be brought in. This allows unexpected associations to be brought in, even though they are not on the surface at all.

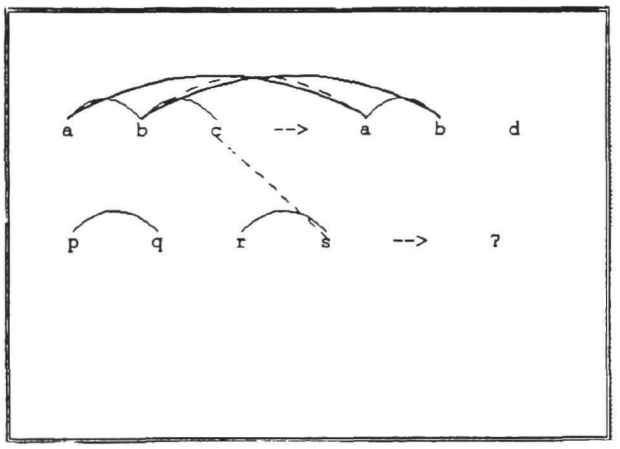
The following is a run of the program on the problem "If **abc** ---> **abd**, then **pqrs** ---> ?" This run produced the answer **pqrt**. This answer is almost always produced by the program, although on rare occasions it produces the rigid answer **pqrd**.



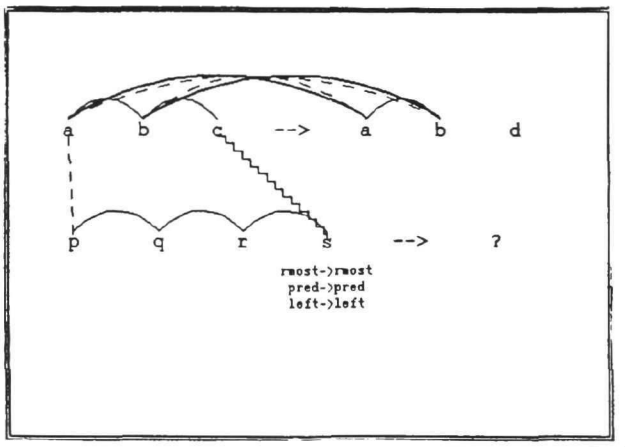
1. The program is presented with the three strings.



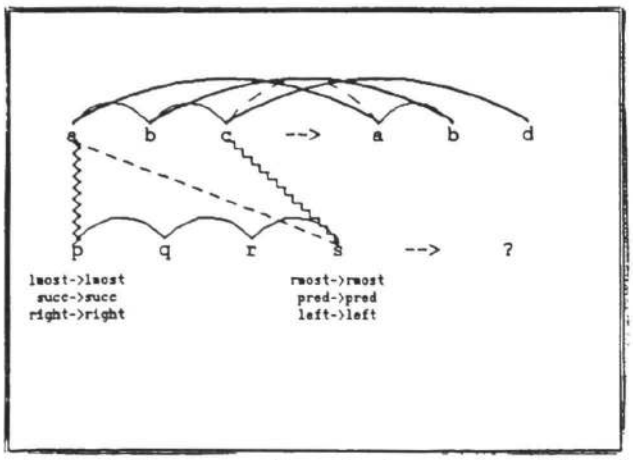
2. Tentative correspondences between letters in the two top strings are being considered (dashed arcs). A successor/predecessor relation has been noticed between the **A** and the **B** in **abd** (solid arc).



3. Correspondences between the two **A**'s and between the two **B**'s have been built (solid arcs). A competing correspondence between the **B** in **abc** and the **A** in **abd** is tentatively being considered. More successor/predecessor relations have been noticed. A tentative correspondence between the **C** in **abc** and the **S** in **pqrs** is being considered.

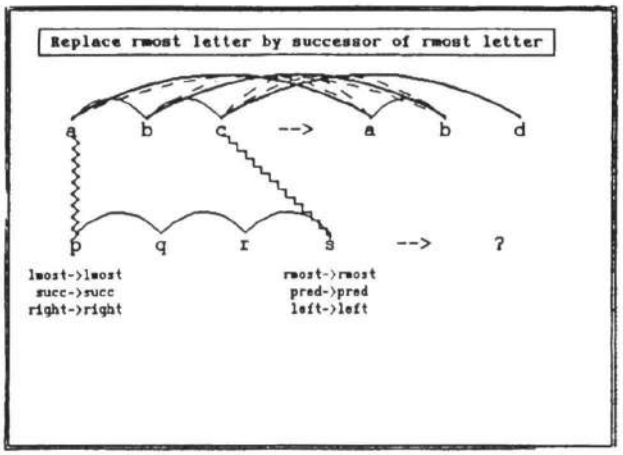


4. The **C-S** correspondence has been built (jagged line) and at the bottom are listed the three trivial slippages underlying it. The slippage "rmost --> rmost" means that both letters are rightmost in their respective strings. The slippages "pred --> pred" and "left --> left" indicate that each letter's left neighbor is its predecessor. A correspondence between the **A** and the **P** is being considered, which would be compatible with the **C-S** correspondence. Meanwhile, alternative correspondences on the top line are being considered.

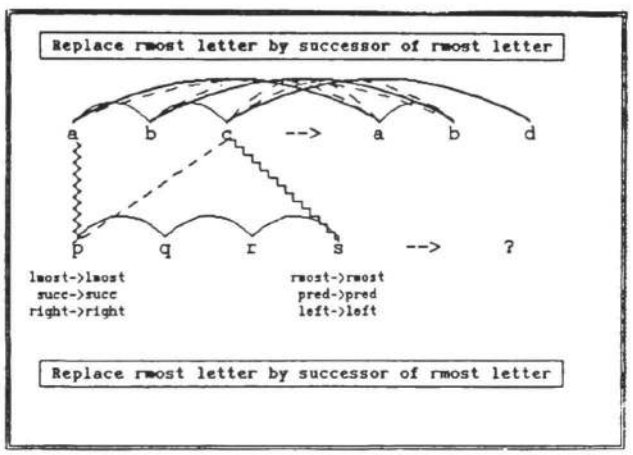


5. The **A-P** correspondence has been built, and there is a competing tentative correspondence between **A** and **S**, based on the notion that the **A** is leftmost and the **S** rightmost, two concepts close enough in the Slipnet to allow a correspondence to be considered, but, it turns out, not close enough in this context for that correspondence to compete. Incompatible tentative correspondences can coexist, but incompatible genuine correspondences cannot, so various incompatible sets of correspondences compete on the basis of strength.

The mapping between **abc** and **abd** is complete, but an alternative correspondence is being considered.

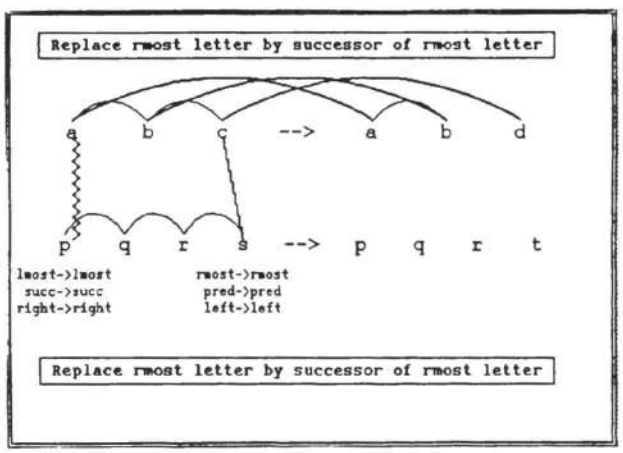


6. The tentative **A-P** correspondence was weak, and died. A rule describing the change from **abc** to **abd** has been written at the top. Copycat currently is limited to situations where just one object is changed, so rules are made by filling in a template of form "Replace _____ by _____ ". The rule-building codelet finds the changed object, and probabilistically chooses a description of it, preferring salient and abstract ones. E.g., "rightmost letter" is more abstract than "instance of **C**", though the latter is occasionally chosen. Likewise, a description of the corresponding object in the second string is probabilistically chosen.



7. The rule has been "translated" for use on the target string, and appears at the bottom. The slippages underlying the correspondences are used as translation rules; in this problem there is nothing to translate, since "rightmost", "successor", and "letter" play the same role in **pqrs** as in **abc**. As will be seen, however, this is not so when the target is **ssrrqqpp**.

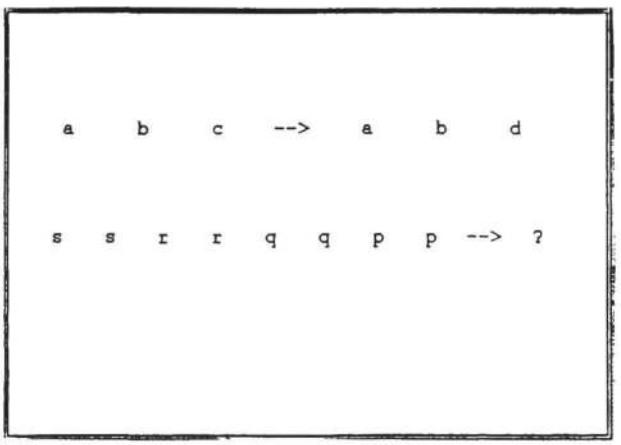
Even though the rule has been translated, alternatives to the mappings are still being considered. If any of these alternatives were to succeed, then the current rule would be discarded and a new one would have to be constructed.



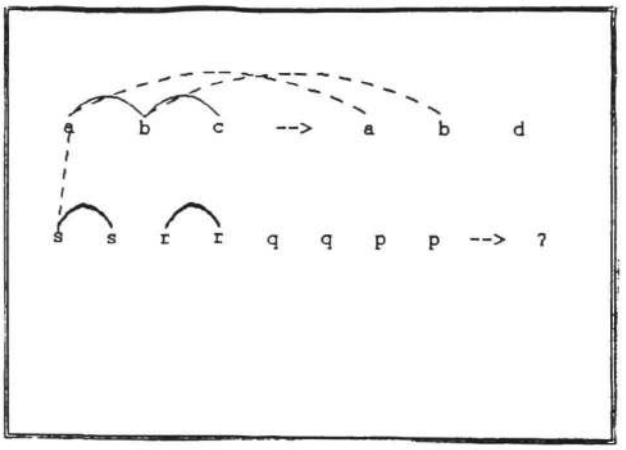
8. The program has used the translated rule to create an answer: **pqrt**.

Notice that no correspondence was ever made between the **B** in **abc** and anything in **pqrs**. This reflects the fact that in an analogy between two situations, not every aspect of each situation has to be mapped. In the case of these miniature situations, there is no good counterpart for **B** in **pqrs**.

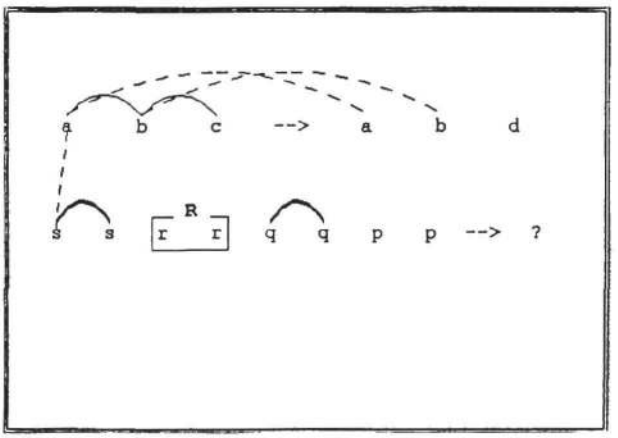
Next, the program is run on the problem "If **abc** ---> **abd**, then **ssrrqqpp** ---> ?" This run produced the answer **ssrrqqoo**, but note that the answer **ttrrqqpp** is also produced quite often, and on rare occasions rigid answers such as **ssrrqqpq** and **ssrrqqpd** are produced.



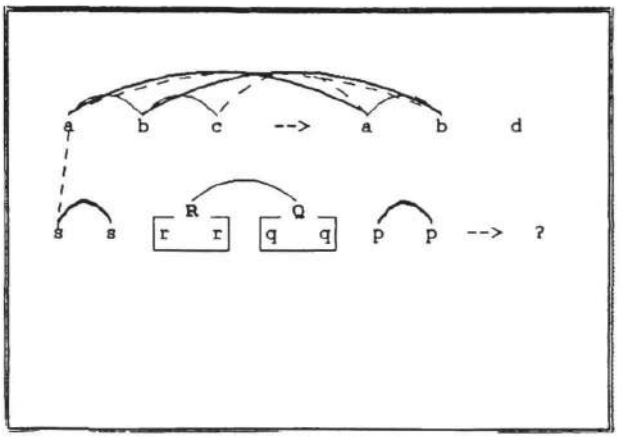
1. The program is presented with the three strings.



2. Successor/predecessor relations (light solid arcs) and sameness relations (dark solid arcs) between letters are beginning to be noticed, and some tentative correspondences have been set up between **abc** and **abd**. In addition, a tentative correspondence has been made between the **A** in **abc** and the leftmost **S** in **ssrrqqpp**.

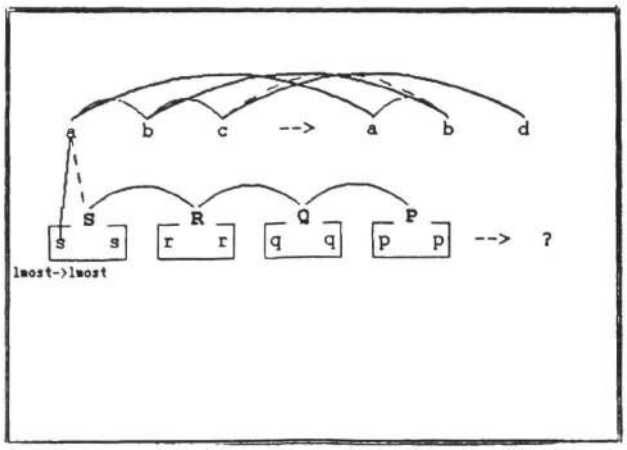


3. A group is formed out of the two **R**'s bonded by a sameness relation. The group is represented by a parameter-letter **R** (the boldface **R** appearing above the group). A parameter-letter acts much like a letter, but exists at a more abstract level.

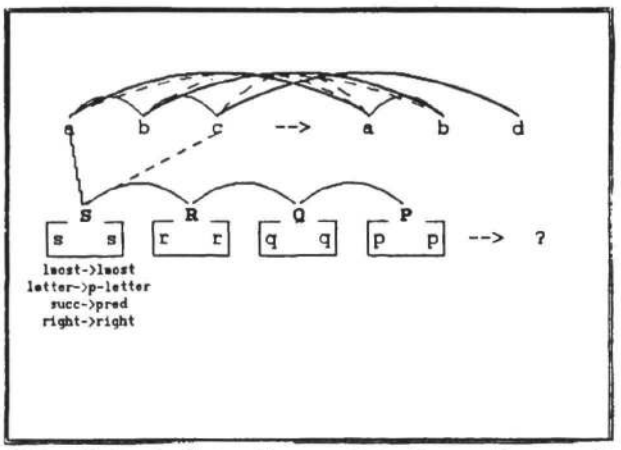


4. The group of **Q**'s has been perceived, and is characterized by the parameter-letter **Q**. This allows a successor/predecessor relation to be noticed between the parameter-letters **R** and **Q**.

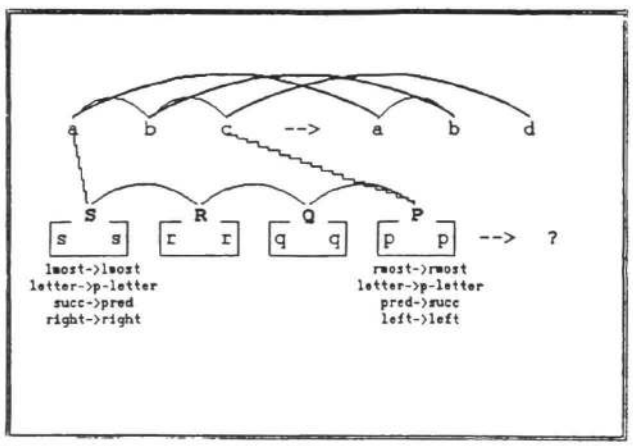
Some correspondences between **abc** and **abd** have been built, and others are being considered.



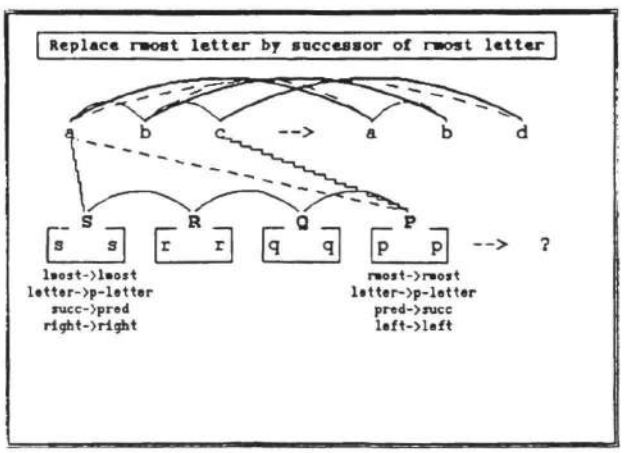
5. All the groups in **ssrrqqpp** have been noticed, and all the successor/predecessor relations at the parameter-letter level have been noticed. (The already-perceived successor/predecessor and sameness relations at the letter level are still present, but are suppressed from the graphics.) Also, a mapping between **abc** and **abd** has been completed. In addition, a correspondence has been built between the **A** in **abc** and the leftmost letter **S** in **ssrrqqpp** (jagged line). Its only (trivial) slippage is "leftmost --> leftmost". However, a rival correspondence is being considered between the **A** and the parameter-letter **S**.



6. In a fight, the latter destroyed the former, because it is supported by more slippages (4 vs. 1), meaning more similarities are being taken into account, and because they involve more abstract concepts (like "parameter-letter"), meaning <u>deeper</u> similarities are being taken into account. Its slippages tell us: "leftmost" plays the same role in both strings; "letter" in **abc** corresponds to "parameter-letter" in **ssrrqqpp**; "successor" to "predecessor", and "right" to "right", since **abc** increases alphabetically to the right, while **ssrrqqpp** decreases (at the parameter-letter level). Meanwhile, "diagonal" competition has appeared.

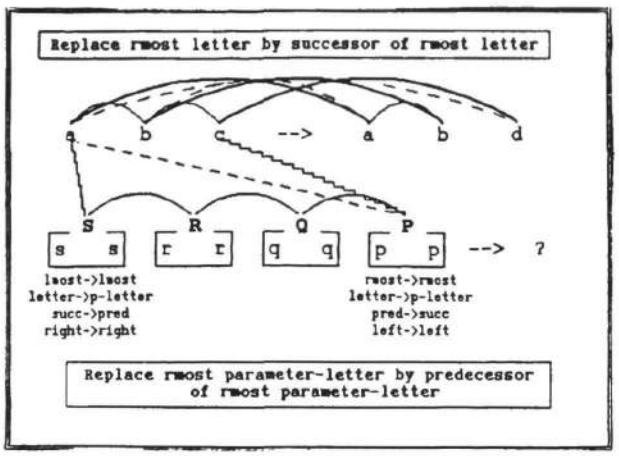


7. The diagonal competition is gone. In contrast to the previous problem, diagonal correspondences (representing the view that the strings have the same alphabetic order, but opposite spatial directions) are as strong as the vertical correspondences (representing the view that the strings are read in the same spatial direction, but with alphabetic direction reversed). Both these views are reasonable, though only one can exist at a time. The diagonal mapping lost only because of an unlucky throw of the dice. The compatible correspondence (from **C** to parameter-letter **P**) has been built, completing the vertical mapping.

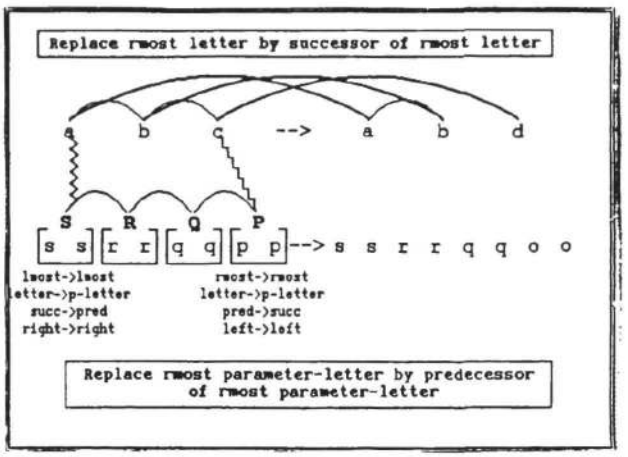


8. A rule expressing the change in the first line has been constructed (the very same one as was constructed in the first problem). But note that if it were applied directly to **ssrrqqpp**, it would yield the rigid and unappealing answer **ssrrqqpq**; therefore, it will have to be translated.

Another attempt is being made to construct a diagonal mapping (here between the **A** and the parameter-letter **P**). If it were successful, then the two existing correspondences would be destroyed.



9. The rule has been translated according to the translation recipes embodied in the slippages underlying the correspondences.



10. The answer **ssrrqqoo** has been created according to the translated rule. If the diagonal correspondences had won, the rule would have been "Replace the leftmost parameter-letter by the successor of the leftmost parameter-letter", yielding answer **ttrrqqpp**.

These runs demonstrate Copycat's current capabilities; solving the subtler analogy problems mentioned above will require some additions to the architecture. Our plans for future work are discussed in Hofstadter, Mitchell, & French (1987) and in Mitchell (1988). We are also testing the generality of our approach by using similar architectures in different domains (Hofstadter, Mitchell, & French, 1987; Meredith, 1986). Comparisons between Copycat and other models of analogy-making (especially work by Gentner, 1983 and Falkenhainer et al., 1986, and by Holyoak and Thagard, 1987) are given in Hofstadter, Mitchell, & French (1987) and in Mitchell (1988).

Acknowledgements

We thank Robert French for many important contributions to this project. This research has been supported by a grant from the University of Michigan, a grant from Mitchell Kapor, Ellen Poss, and the Lotus Development Corporation, a grant from Apple Computer, Inc., and grant DCR 8410409 from the National Science Foundation.

References

[1] Erman, L.D., F. Hayes-Roth, V. R. Lesser, & D. Raj Reddy (1980). The Hearsay-II speech-understanding system: Integrating knowledge to resolve uncertainty. Computing Surveys, 12(2), 213-253.

[2] Falkenhainer, Brian, Kenneth D. Forbus, & Dedre Gentner (1986). The structure-mapping engine. In Proceedings of the American Association for Artificial Intelligence. Los Altos, CA: Morgan Kaufmann.

[3] Gentner, Dedre (1983). Structure-mapping: A theoretical framework for analogy. Cognitive Science, 7(2).

[4] Hofstadter, Douglas R. (1984b). The Copycat project: An experiment in nondeterminism and creative analogies (AI Memo #755). Cambridge, MA: MIT AI Laboratory.

[5] Hofstadter, Douglas R. (1985). Analogies and roles in human and machine thinking. In Metamagical Themas (pp. 547-603). New York: Basic Books.

[6] Hofstadter, Douglas R., Melanie Mitchell, & Robert French (1987). Fluid concepts and creative analogies: A theory and its computer implementation. Technical Report 10, Cognitive Science and Machine Intelligence Laboratory, University of Michigan, Ann Arbor, Michigan.

[7] Hofstadter, Douglas & Melanie Mitchell (1988). Concepts, analogies, and creativity. To appear in Proceedings of the Canadian Society for Computational Studies of Intelligence. Edmonton: Univ. of Alberta.

[8] Holyoak, Keith J. & Paul Thagard (1987). Analogical mapping by constraint satisfaction. Manuscript submitted for publication.

[9] Meredith, Marsha J. (1986). Seek-Whence: A model of pattern perception. Unpublished doctoral dissertation, Indiana University, Computer Science Department, Bloomington, Indiana.

[10] Mitchell, Melanie (1988). A computer model of analogical thought. Unpublished thesis proposal, University of Michigan, Computer Science Department, Ann Arbor, Michigan.

Direct Inferences in a Connectionist Knowledge Structure

S. C. Hollbach
University of Rochester

Abstract

A model of human cognition is proposed in which all concept properties are context dependent. Concepts are comprised of multiple facets, each motivated by a different functional property. A connectionist implementation is presented in which conceptual modification yields the 'direct inferences' implicit in the structure of a knowledge base.

Introduction

When a noun is modified by a descriptive adjective, the result is often a significant modification of the original concept denoted by the noun. For example, while a peach is soft, juicy and tangy, a green peach is hard, dry and bitter. Clearly, the adjective 'green' when applied to a peach conveys more than merely the colour.

This paper advancces a computational model of conceptual modification that captures the 'direct inferences' arising from property correlations. The domain of this investigation is concrete nouns and their attendant descriptive adjectives. Each noun denotes a concept, where a concept is represented by a structured collection of properties and values, indexed by function. For example, an apple viewed as food brings different properties to mind than an apple viewed as a projectile. Ultimately it is the goals and plans of the agent that determine how an object is thought of: a hungry agent thinks of apples differently than an angry one. Thus the functional properties of an object provide the context for interpretation, in that they select only the currently relevant facet of the complete description.

We have built a connectionist implementation of the functional context-sensitive model of category representation. The system runs on a Sun Workstation as an application of the Rochester Connectionist Simulator [Goddard 1987], the results of which are visible in iconic form thanks to the Graphics Interface [Lynne 1987]. The system uses an extensive knowledge base of categories and their interrelations to draw direct inferences about modified categories, answer queries about object properties, and model property dominance effects [Whitney 1986, Tabossi 1986].

The Structure of Knowledge

This investigation focusses on the mental representation of physical objects. The building block of these mental representations are *categories,* classifications of physical objects sharing one or more common properties. A *property* is a set of descriptors applicable to a physical object, where the elements of the set are *property values.* We classify properties into three groups, perceptual, constitutive and functional. Perceptual properties pertain to the five senses, functional properties relate to an object's usefulness by humans, and constitutive properties are in some sense the definitional properties of a category, often expressed in terms of genetics, compositional makeup and so on. Functional properties play a special role in category representation, supplying as they do the various perspectives from

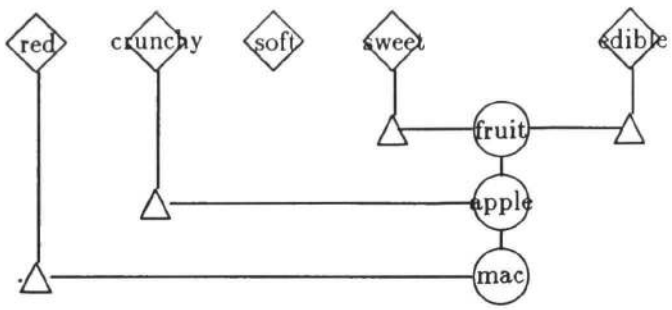


Figure 1: Circles represent categories, diamonds property values and triangles binder nodes associating categories with their attendant values. Lines do not represent direct links, but rather indirect connections mediated by subnet structures of varying complexity.

which the category can be viewed. For example, the *is edible* property provides the focus of relevance for the *tangy* and *has seeds* properties of apples.

Categories, in addition to having a comples internal structure, are related to one another in a hierarchical subsumption taxonomy. (A familiar example of such structuring is ontological knowledge, the ordering of natural kinds according to common biological characteristics.) The links in the taxonomy represent subsumption relations between categories. Thus to some degree a lower level category participates in the higher level category. The form this participation takes differs depending on whether one is looking up or down the taxonomy. All the properties and values possessed by the higher level category are also possessed by the lower level one. And for each property or value possessed by a lower level category it is true of the higher level category that there exists an element of that set having that property or value. For example, since all things have colour as a property, all apples must also have a colour. Furthermore there exists a red (or green or yellow) apple, by virtue of the colour values associated with the various apple varieties.

The connectionist implementation of this cognitive model follows the 'localist' paradigm of Feldman and Ballard [1982]. Each category is represented by a single exemplar object. Each object, property and value is represented by a distinct (named) network node. All relations between these nodes are captured in separate subnets of regular structure, allowing the network to be compiled from a series of high level input language statements. Concepts are represented as patterns of activity over all the nodes in the network. 'Thoughts' are formed in the network by keying in activation on a noun and (optionally) adjectives, and allowing the simulation to run a few steps to permit these activations to propagate fully. Activation flows out from the noun denoting the category to all relevant properties and values. Relevance is determined by context, or more specifically, by the currently active functional property of the category. Each category has associated with it a default context or facet; for example, the default view of 'apple' is *edible*. So when the noun is activated in isolation, the system responds by selectively activating its defining properties and values with respect to its default context. For example, Figure 1 depicts the graphics display of certain key elements of the network after keying in activation on the 'apple' node and allowing the simulation to run a few steps to stability. The selective effects of context mean that only a subset of all possible property values of the concept are active at one time, although a given property value can participate in any number of facets.

Direct Inferences

A major feature of the connectionist knowledge base is its dynamic nature. Rather than having the knowledge encoded in a purely passive (declarative) format requiring a distinct reasoning component to interpret and apply it, the knowledge encoded in a connectionist network incorporates several simple forms of inferencing directly in the structure of the network. These direct inferences, that is, inferences not requiring an interpreter but contained entirely within the terminological component, can be either *mediated* or *immediate*. Immediate inferences are drawn about object properties at the level of the object itself, while mediated inferences involve property inheritance. Mediated inferences can be drawn either from more general knowledge, or, if the information is not available at a higher level, a weaker anwer can be derived from more specific knowledge.

Both forms of direct inference, mediated and immediate, arise from this fundamental mode of operation of the network, as demonstrated in Figure 1. One of the many immediate inferences drawn about apples is the fact that they are crunchy; one of the mediated inferences is the fact that they are edible. A further inference, as to the existence of red apples, is obtained by ranging down the hierarchy, rather than up as is customary.

Property Queries

Property queries take the general form "does (modified) category x have property value y?". Phrased more naturally, this becomes "are y's x?" or "do y's have x's?". Of course, given that each category is represented by a single exemplar, a more accurate portrayal of the query forms would be to say "is a y x?" or "does a y have x?", for example, "is a black bird large?" or "does a red apple have seeds?". Each property value is represented by two nodes, one corresponding to the adjective as a category modifier, the other corresponding to a query on that property value. So to pose a query to the system, the user activates the adjective and noun forming the target category, thus invoking the fundamental mode of operation of the network, namely, the drawing of direct inferences. The queried property value is then keyed in on the query-specific twin of the property value node.

There are five possible responses to a query: a 'yes' or 'no' in context, a 'yes' or 'no' out of context, and 'category error'. An 'yes' in context occurs when the property value is an element of the set characteristic of the current facet. A 'no' in context occurs when a modifier negates the queried value, as in "is a green apple red?". A 'yes' out of context is reported when a shift of context is required to answer in the affirmative, as in "are sweet apples easily thrown?". An answer of 'no' out of context results when the property value is not associated with the category in any context, although other values of that property are, as in the query "are small apples purple?". Answers out of context take a little longer than answers in context, since context shifting takes time. A category error occurs when the property associated with the value is not a property of the category ("are ideas purple?"). Category errors are interesting not only for their possible role in cognitive development [Keil 1979], but also for the fact that when they occur in conversation it is generally to signal a metaphor. We are currently in the process of extending the model to account for metaphoric interpretation of such category errors.

Once the query has been keyed in, the simulation is run to a point where either an answer of 'yes' or 'no' in context or a report of a category error would be detected by the system. If neither condition exists, alternate facets are explored in parallel until either an answer of 'yes' out of context is reported or the possibilities are exhausted, resulting in a 'no' (out of context).

Property Dominance Effects

In addition to participating in a subsumption taxonomy, the mental representation of an object has a complex internal structure. The design of this internal structure is based on the premise that all object properties are context dependent. This idea arose from the debate between Whitney [1986] and Tabossi [1986] over the problem of lexical access modelling, or the question of whether the meaning of an ambiguous word is selected at the lexical access stage or interpreted later. Whitney presents related results concerning the semantic access of unambiguous words in support of the multiple access model (akin to the delayed interpretation model for ambiguous words). In Whitney's work, all the properties germane to a concept (as denoted by a concrete noun) are accessed or primed in parallel, by mention of the noun, regardless of any bias built into the sentential context. The effect of the bias, to promote some properties to prominence and inhibit others, is only visible several hundred milliseconds after inital mention of the nouns, and must thus be occurring at a later (post-lexical) processing stage. Tabossi, on the other hand, contends that the stimuli used in Whitney's work are too neutral with respect to the target concept to induce any significant bias, and presents results to support the competing notion of selective access, or lexical level biasing and inhibition of properties.

The question of whether or not lexical access is context sensitive is still an open one. There is agreement in the literature, however, on the fact that certain concept properties are correlated, both positively and negatively [Malt and Smith 1984], and that context is used to decide which of a number of competing property associations or coalitions should be permitted to dominate [Cohen and Murphy 1984]. As Tabossi points out, ice is both hard and cold, yet a sentence like "The bartender served the drinks with ice" that primes the property value 'cold' also inhibits the value 'hard' and vice versa, while a neutral sentence neither primes nor inhibits either property.

These results are consistent with the model advanced in this work, in that competing contexts are mutually exclusive, neutral contexts are unrelated, and reinforcing contexts are mutually excitatory. Dominance effects are modelled by asymmetric link weights between the two competing facets, permitting a high dominance property (*eg.* ice temperature) to exhibit stronger effects than a low dominance one (*eg.* ice hardness).

A fundamental assumption underlying this work is that categories, as mental constructs of active agents, are inseparably linked with the agent's planning goals. These goals or situational contexts are so influential on the mental structure of categories that a category is meaningless when out of context. Since categories (indeed, all ideas) are by definition meaningful, they must carry with them a default context to supply meaning in the absence of other information. Very often, particularly for physical objects, this default context is simply visual recognition. When the word 'apple' is spoken, a mental image is conjured up of the visual appearance of an apple. If the agent is hungry at the time, the apple's taste

might spring to mind. If the agent is angry, its properties as a handy projectile might leap into significance. And so on. Thus while a category appears stable to the agent, since the same basic set of properties and values are being drawn on at all times, the structure is actually dynamic, shaped by context.

Contexts interact amongst themselves in different ways than do categories. Where categories can combine with each other in an arbitrarily complex fashion, contexts afford less latitude. A context is nothing more than a particular way of looking at a category. Two perspectives on a category are either the same or different. So contexts can be either mutually supportive or mutually inhibitory, depending on whether they are compatible or not. For example, if the agent's goal is to eat an apple, he must first locate one visually. Thus the *edible* and *visual-id* contexts are compatible. On the other hand, if the agent decides to throw the apple at a passing car, any thought of eating it will be suppressed, as throwing and eating are mutually inhibitory contexts.[1]

The general characteristics of the model derive partly from the psychological model described above and partly from properties of connectionist models. Categories are represented by the various patterns of activity over the set of properties and values associated with that category. Each distinctive pattern is characteristic of a different context or goal. There is a default context associated with each category. If in the process of specifying the structure of the knowledge base the user fails to name a default context for a category, the 'visual-identification' context is used, since this is generally appropriate for the chosen domain of physical objects.

Property values can be either *neutral* or *biasing*. A neutral value displays no strong correlation with one context over any other, while a biasing value is characteristic of only one context. For example, the modifier 'red' is neutral in the phrase 'red pillow' but biasing in the phrase 'red rose', raising as it does visions of romance and long-stemmed floral offerings. Thus context can be established implicitly by mentioning a biasing property value; it can also be established explicitly, by turning on the context node by hand. Not all property values are biasing, or, more accurately, not many property values are sufficiently biasing to override the currently active or the default context in favor of another. The values biased toward the context can be guessed at with greater confidence than the more neutral ones, although there is a slight bias built into neutral values. In fact, the neutral/biasing distinction is not a very good one, as it represents the attempt to quantify a gradual change. A more accurate characterization would be to speak of strong, moderate, weak and negligible biases.

Results of running the simulation are shown in Figures 2, 3 and 4. Shown are the iconic representations of individual network nodes. Activation was keyed on the phrase 'expensive diamond' and the simulation allowed to run to stability. Figure 2 shows this initial state of the network. Figure 3 shows the network in an intermediate state shortly after changing contexts from the *adornment* aspect of diamonds to the *industrial* aspect, achieved by shifting the external activation of 'expensive' over to 'hard'. That is, the system is being forced to consider the phrase 'hard diamond' after being primed with the phrase 'expensive diamond'. As Tabossi's property dominance studies predict, there is a significant latency period between presentation of the stimulus and recognition of its appropriateness, as shown

[1]This version of events is admittedly simplistic, but it suffices for the problem at hand.

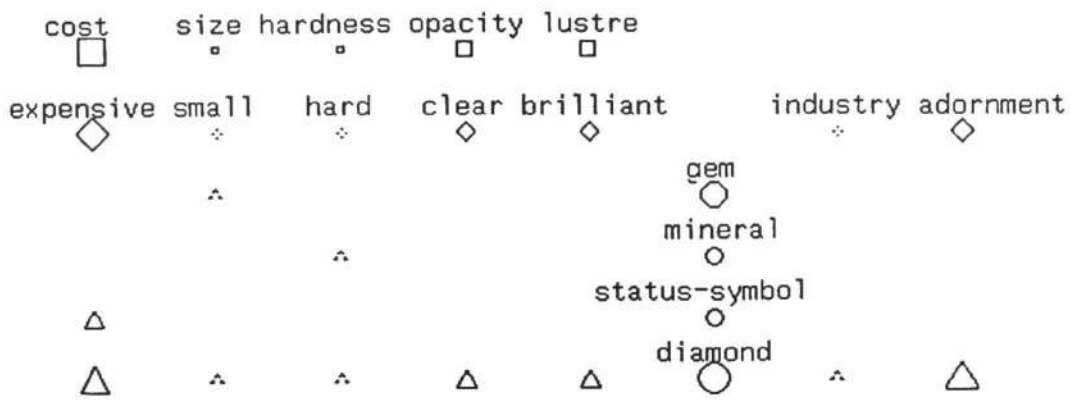


Figure 2: Graphics Interface depiction of changings contexts.

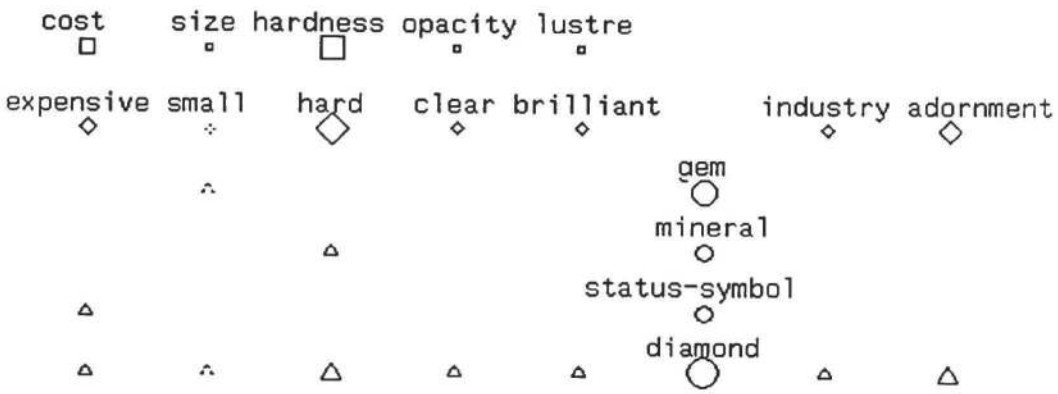


Figure 3:

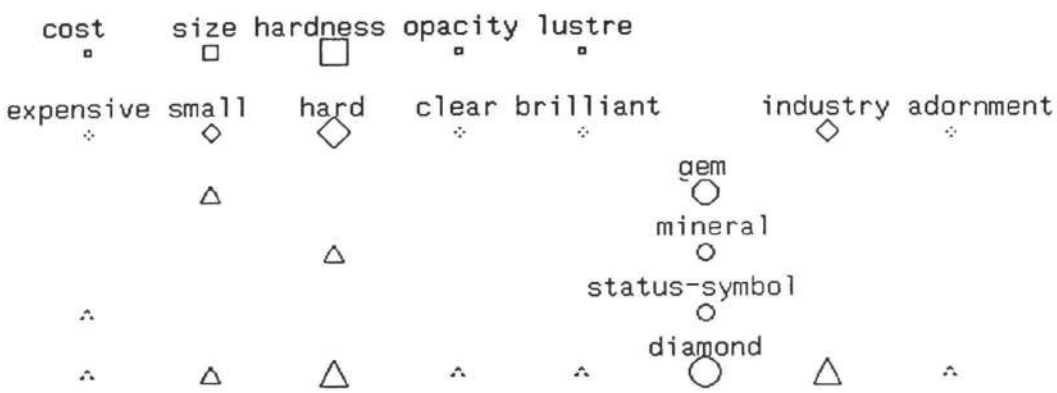


Figure 4:

in Figure 4. The details of the implementation, including a description of a connectionist interpreter that translates statements in a high level language into a structured network, are given in [Hollbach 1988].

Summary and Conclusions

This research advances a context sensitive model for conceptual modification and uses it to capture not only property dominance effects but also direct inferences, both immediate and mediated. Concepts are denoted by concrete nouns, optionally modified by one or more descriptive adjectives. The agent's current plans and goals supply the relevance criteria for focussing on a coherent subset of a given object's disparate properties and values. Goals are represented simply by a concept's functional properties. A characteristic use of an object will dominate uncommon ones, as will all perceptual and constitutive properties associated with that functional property, leading to property dominance effects. The interproperty assocations yield immediate inferences, and property inheritance gives rise to mediated inferences. The connectionist implementation of this model operates as a question-answering system, permitting the user to pose queries about the various properties of a concept.

Acknowledgements

Thanks to Jerry Feldman, and to Gary Dell and Josh Tenenberg for invaluable comments.

References

[Cohen and Murphy 1984] Benjamin Cohen and Gregory L. Murphy, "Models of Concepts", *Cognitive Science*, 8:27–58, 1984.

[Feldman and Ballard 1982] Jerome A. Feldman and Dana H. Ballard, "Connectionist Models and Their Properties", *Cognitive Science*, 6:205–254, 1982.

[Goddard 1987] Nigel Goddard, "The Rochester Connectionist Simulator User Manual", Technical Report, Computer Science Department, University of Rochester, April 1987.

[Hollbach 1988] Susan C. Hollbach, Automatic compilation of a connectionist knowledge base, 1988, submitted to AAAI'88.

[Keil 1979] Frank C. Keil, *Semantic and Conceptual Development An Ontological Perspective*, Harvard University Press, Cambridge, Mass., 1979.

[Lynne 1987] Kenton Lynne, "Graphics Interface to the Rochester Connectionist Simulator", Technical Report, Computer Science Department, University of Rochester., April 1987.

[Malt and Smith 1984] Barbara C. Malt and Edward E. Smith, "Correlated Properties in Natural Categories", *Journal of Verbal Learning and Verbal Behaviour*, 23:250–269, 1984.

[Tabossi 1986] Patrizia Tabossi, Effects of context on the immediate interpretation of unambiguous nouns, 1986, Universita di Bologna, Italy. Unpublished.

[Whitney 1986] Paul Whitney, "Processing category terms in context: Instantiations as inferences", *Memory & Cognition*, 14(1):39–48, 1986.

Problem Solving is What You Do When You Don't Know What to Do

Laura K. Kochevar
Department of Psychology

Paul E. Johnson
Carlson School of Management
University of Minnesota

Experts make it look easy. They perform their tasks efficiently, and usually, correctly. Tasks which novices perform only at the expense of a great deal of time and effort are accomplished almost immediately, seemingly intuitively by the expert. In developing these skills individuals typically lose the ability to tell us what they do (Johnson, 1983). The fluency of expert performance and the tacit nature of expert knowledge can be understood if viewed as the result of adaptations which facilitate performance in a specific task environment.

Adaptations can be described in terms of environmental specificity, goal relevance, and function. The specific adaptations experts develop are collectively referred to as "knowledge". An agent's knowledge can be described in terms of how the agent knows what can be done to solve a problem. In this case the goal is problem solution, the environmental information is "what can be done" and the function is "knowing". In most theoretical descriptions one or more of these aspects of performance remain implicit. For example, saying "the giraffe uses its long neck to reach food" focuses on the relation of the adaptation to goal directed function -- leaving environmental specificity implicit. Theories of expert performance usually address the issue of how the expert knows how to solve the problem -- also leaving environmental specificity implicit.

Of course, in order for the giraffe's long neck to be facilitate feeding, the giraffe's environment must be structured so that food exists in high places. If knowledge is adaptive then the problem environment must also be structured so that knowing facilitates goal attainment. If the structure of the environment does not specify goal relevant actions then there is nothing to adapt to, nothing to know.

Not all animals eat "leaves in high places". Anteaters have long tongues rather than long necks. The anteaters tongue is a different adaptation to a different feeding environment. Both the anteater's tongue and the giraffe's neck function as "reachers" to accomplish the "feeding" goal. The different adaptations correspond to different environmental

structures. If we were to develop a model of "food reachers" without concern for the structure of "food" we would not readily propose two such radically different models.

The critical question for those constructing models of cognitive processes is "what constitutes food for thought?" We propose that the expert and the novice function in very different task environments. Expert level performance is the result of adaptations which allow extraction of information from the environment which differs from information available to naive performers. It is our assertion that current cognitive theories fail to capture the qualitative difference between expert and novice level performance because they assume individuals function within the same nominal environment.

An environment contains information which is structured in a way that indicates what you can do with it (Gibson, 1979/1986). Most environments are informationally "rich". That is, there is quite a lot you can do with them, hence, they have many structural descriptions. Even the very restricted "environment" which we label "tree" presents various information structures which specify "climb up-able", "obstacle", "hide behind-able", etc. These information structures indicate the real physical characteristics of objects and events that make them useful in task performance. A physician is able to diagnose a patient's disorder because the disease produces physiological deficits which in turn produce symptoms. In this sense, the environment is objective -- the relationship between symptoms and disease reliably exists apart from an observer. The task environment however, is subjective in that it consists of the information meaningful to a particular individual seeking to attain a particular goal.

The environment can be described in terms of its "surface structure" or nominal features. The relationships among features can also function as sources of information. Certain sets of features and feature relationships specify goal relevant characteristics of the environment. Given a fixed goal, and a fixed behavioral potential, there is some set of features and feature relationships which specifies the optimal goal directed behavior. This is the "deep structure" of the task environment.

Experts perform by selecting from the environment information specifying the "deep structure" of the domain. Novices are only capable of accessing the "surface structure". Since novices do not access task relevant feature relationships they must mentally combine surface level information or "deduce" task solution. To the extent that the novice's reasoning consists of invoking processes functionally similar to

environmental structures, the novice will be able to perform the task. Experts need not reason. They can behave intuitively because they can access relational information from the environment.

The distinction between expert and novice level performance is paralleled by the nature of the knowledge underlying each. The expert has task specific adaptations which correspond to the environmental structure related to task performance. "Expertise" is this body of operative knowledge. The novice's adaptations do not allow direct knowledge of the information which adequately specifies performance. Therefore, the novice's operative knowledge must be supplemented by procedural knowledge. The procedural knowledge is the set of cognitive processes which manipulate operative knowledge to derive predictions about the environment.

The fluent, implicit nature of much of an individual's operative knowledge presents a serious problem in the study of expertise (Berry, 1987). Most investigations of expert performance attempt to discover the structure of operative knowledge by placing the expert in a task environment where his or her adaptations fail (Simon, 1969). This forces the expert to invoke procedural knowledge. Although an explicit behavior trace can be generated by this method we cannot be sure that the knowledge observed is similar to that normally used.

Most cognitive theories resolve the problem of implicit knowledge by assuming intrinsic, combinatorial semantics (Fodor and Pylyshyn, 1988). The meaning of a particular representation is assumed to be reducible to the meaning of its constituent features. Since meaning at any level of description can be derived from meaning at another level, it can be assumed that the explicit knowledge trace produced by an expert individual performing in a novel environment is functionally equivalent to the underlying expertise.

In the model we have presented the semantics are environment referenced. This allows varying degrees of relationships between knowledge levels. Given that knowledge is viewed as an adaptation to the environment, and what one knows about the environment is what one can do with it, we can assess the structure of implicit knowledge directly by studying the relationship between environmental information structures and behavior.

We have applied this model to understanding the knowledge used by an expert in the diagnosis of congenital heart diseases. We expected that by analyzing the history of a specific individual's interactions with the environment we could describe this individual's functional environment. Eventual outcome measures were also available, allowing us to describe a

hypothetical optimal knowledge structure. Since multiple individuals treated each patient we could construct a knowledge description of a composite "other"; in this case, a composite novice.

METHOD

For this analysis,15 patient charts were chosen from the records of the University of Minnesota Heart Hospital. The same expert physician had diagnosed all 15 cases. Each case had also been diagnosed by one or more less experience physicians (novices). After physician diagnosis each patient's actual disease state was determined by cardiac catheterization. Each chart contains the record of symptoms noted, the diagnosis each physician assigned to the case, and the actual underlying disease state. There were three major diseases represented: transposition of the great vessels (TGV), ventricular septal defect (VSD) and atrial septal defect (ASD). It is possible for any patient to have either disease in isolation, both diseases simultaneously, or neither disease. Each of these 15 cases had at least one disease.

Of the three general types of analytical methods available for describing patterns of symptom variation (clustering, multidimensional scaling, and factor analysis); factor analysis is conceptually most similar to the theory we have presented. Unlike clustering methods, factor analysis provides ways to use one symptom in combination with others to indicate more than one underlying pathology. For example, while the combination of blood pressure and age may indicate a cardiac abnormality; the combination of weight and age may signal that the patient was born prematurely or is suffering from malnutrition. Multidimensional scaling techniques are based on dissimilarities among cases while factor analysis methods identify patterns of symptom covariance. The use of factor analysis allows us to identify factors which indicate a particular disease without implying the absence of other disorders.

Factor analytic methods are normally applied to a limited number of variables measured on a large numbers of cases. The goal is to explain relationships among variables in terms of fewer, more general, hypothetical constructs. In this example we are using the technique to mathematically describe this particular sample of 15 cases. We are not concerned at this point about stability of factor loadings or generalizability to other samples. Explanation and data reduction are accomplished by determining which symptom patterns are related to disease state and diagnosis.

What the physician knows about the environment is how symptom structures are related to physical defects. To describe this knowledge

three separate analyses are performed. Two of these relate the expert or novice physician's diagnosis of each case to available symptom patterns. The third uses disease state as determined by catheterization to show which of the available symptom patterns are related to actual pathology. In each analysis the diagnosis or disease of each case is treated as an additional "symptom". All other symptom values are constant across analyses for each case. Any resultant differences in factor structure are due to diagnostic differences and reflect the relationship of symptom patterns to diagnosis and disease.

The factor analyses only indicate which symptom interrelationships covary with diagnostic behavior or disease. They do not tell us what information was actually used. Nor does the analysis indicate what the information means to the physician.

To address these issues, sample cases were presented to a second expert physician. Case pairs in which the cases differed with respect to a single expert knowledge factor were selected from the original data. In addition, other cases from the original data set were presented paired with a fictitious case constructed as a distortion of the original with respect to a single factor. In all, 6 case pairs were presented for the expert's interpretation. We expected that the expert's behavior would conform to factor based predictions. We also expected that the physician would be able to tell us how cases differed from each other, giving conceptual labels to the empirically derived factors.

RESULTS AND DISCUSSION

Figure 1 shows a simplified view of the optimal, expert, and novice knowledge structures. The ovals represent diseases, the circles represent orthogonal symptom patterns. A circle within an oval is a symptom pattern which specifies the disease. When two diseases are related to the same factor, the factor is bipolar and discriminates between them. A set of important nominal symptoms appears to the right. The symptoms which are associated with factors across descriptions are indicated at the far right. The numbers represent the symptom patterns to which they relate. The lines between the knowledge structures and symptoms indicate additional symptoms contributing to each factor. Each analysis accounted for approximately 90% of the diagnostic or disease variance.

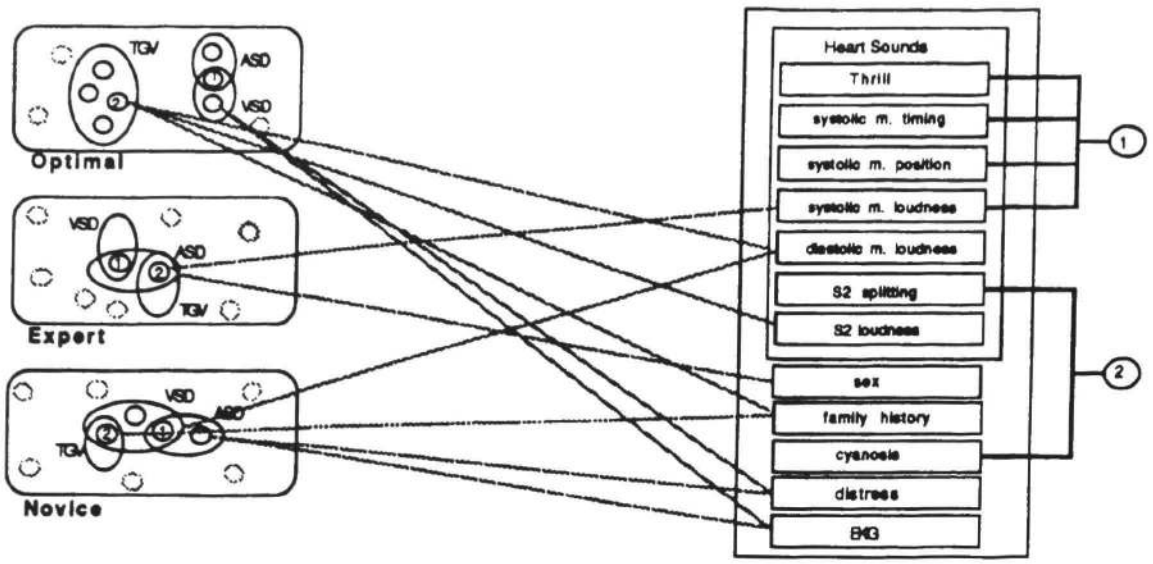


Figure 1 Knowledge structures for optimal diagnosis, expert diagnosis, and composite novice diagnosis.

The optimal knowledge structure shows four factors which specify TGV, two factors which specify ASD and two factors which specify VSD. ASD and VSD share a discriminating factor. TGV is independent of the other diseases. This relationship between two diseases and the independence of the third is consistent with the known physiology of the diseases. ASD and VSD are both variants of "left to right shunts" while TGV is classified as an admixture lesion.

The expert's diagnostic behavior is consistent with two of the optimal factors. In this structure VSD and TGV are specified by independent, single factors. ASD diagnosis is related to both factors. The composite novice description shows a single TGV factor, two ASD factors, and three VSD factors. In this case TGV is independent of ASD and related to VSD. If an individual exhibited a knowledge structure similar to this composite novice we would expect he or she would have to combine information about these four factors to derive a diagnosis. We would expect the expert's behavior to appear more fluent as the expert would only have to combine two factors.

The results suggest that although much of the relevant symptom information is constant across analyses there are some important differences. For example, the factor which discriminates between ASD and VSD for the expert is nearly identical to that in the optimal description. The nature of murmurs which occur early in the cardiac cycle

discriminates between these diseases. The composite novice description shows a similar factor which includes information about diastolic murmurs. This suggests that novices use all murmurs to discriminate between VSD and ASD. We may label the expert factor "systolic turbulence" and the novice factor "turbulent blood flow".

The knowledge descriptions also suggest how information may be used. The information which independently specifies TGV in the optimal description discriminates TGV from ASD for the expert and VSD for the composite novice. The factor which indicates VSD in the optimal description is a combination of negative distress findings and positive EKG data. The corresponding novice factor is composed of positive distress findings and positive EKG data. The novice factor predicts ASD diagnosis. The correlation between the optimal factor and the novice factor is .95. This is an example of a minor feature difference producing a major knowledge difference.

We expected that an expert would perceive variance along a symptom factor in terms of a domain concept (such as "systolic turbulence"). When paired cases were presented to the second test expert we were surprised to find that he was unable to articulate any conceptual difference between cases other than diagnosis. When presented with a pair representing low vs high scores on the ASD-VSD discriminator the expert identified each case appropriately. When presented with cases representing moderately high vs very high scores on the ASD-VSD factor he immediately diagnosed both cases as VSD's, and, pointing to the case with the higher score, said "that one's a large one".

The expert also appropriately diagnosed cases which varied along the other factor. Two of the three pairs presented were diagnosed as ASD's and TGV's. For the third pair, the expected TGV was diagnosed as Truncus, a different admixture lesion. This difference may indicate that the "TGV-ASD" factor is actually an "Admixture-ASD" factor. An alternative explanation is that this expert's knowledge structures differ slightly from those on which the case constructions were based.

These results suggest that the factors obtained from the data provide an accurate description of the knowledge the expert uses to intuitively diagnose a patient's disease. This knowledge provides him with a way of treating large amounts of diagnostically relevant information as one or two chunks. Since none of the individual elements of the novice's knowledge sufficiently species an appropriate diagnosis, the novice must reason out a solution by combining information.

References

Berry, D. C. (1987). The problem of implicit knowledge. Expert systems, 4(3), 144-151.

Fodor, J. A., and Pylyshyn, Z.W. (1988). Connectionism and cognitive architecture: A critical analysis. Cognition, 28(1988), 3-71.

Gibson, J. J. (1986). The ecological approach to visual perception. Hillsdale, New Jersey: Lawrence Erlbaum. (Originally published 1979).

Johnson, P. E. (1983). What kind of expert should a system be?" Journal of medicine and philosophy, 8, 77-97.

Simon, H. A. (1969). The sciences of the artificial. Cambridge, Massachusetts: MIT Press.

Cirrus: Inducing Subject Models From Protocol Data

Bernadette Kowalski and Kurt VanLehn
Departments of Computer Science and Psychology
Carnegie-Mellon University

INTRODUCTION

Verbal protocol data are collected by asking subjects to talk aloud as they solve a problem, and protocol analysis is the process of interpreting such data. Protocol analysis is used routinely by psychologists and other behavior scientists, and more recently, by knowledge engineers who wish to embed the knowledge of human experts in an expert system. However, protocol analysis is notoriously difficult and time comsuming.

Several systems have been developed to aid in protocol analysis. Waterman and Newell (1971, 1973) developed a system that could read the natural langauge of the protocol and produce a formal trace of it (a problem behavior graph). The system, however, did not produce an abstract model of the subject. Bhaskar and Simon (1977) avoided the natural language understanding problem by having a human coder read the protocol and formalize it. The Bhaskar and Simon program, however, was not designed to construct a model of the subject, but instead to test a specific model of thermodynamics problem solving that was built into it. Fisher's (1988) system provides facilities for assigning formal codes to sections of protocols, interrogating a database of codes/sections, performing path analyses, and inferring flow-chart-like models of the subject's behavior.

Cirrus is halfway between the theoretical neutrality of Fisher's program and the theoretical commitment of Bhaskar and Simon's program. It requires the user to provide an underdetermined model, namely, a problem space (Newell & Simon, 1972). A problem space consists mainly of a representation for states and a set of state-change operators. It is not a complete model of the subject's behavior because it does not contain information for selection of operators or goals. The job of Cirrus is to infer from the data the subject's selection strategies. A completely determined, subject-specific model thus consists of the problem space given to Cirrus by the user and the selection strategies constructed by Cirrus to maximize the fit with the subject's protocol data.

Cirrus requires a human to encode the verbal protocol. However, Cirrus can not accept all formal codes, but only those that designate actions (i.e., the application of a primitive state-change operator). Moreover, Cirrus requires that *all* the subject's actions be encoded. A protocol that consists of all and only the actions of the subject is called an *action protocol*. Typically, action protocols are collected by implementing the experimental task on a laboratory computer and saving the subject's keystrokes. A verbal protocol is usually taped at the same time because subjects often make revealing comments about goals, plans, rationales, difficulties, etc.

Cirrus and the human analyst have distinct jobs in analyzing a protocol. The human analyst uncovers the subject's problem space by drawing upon common sense and knowledge of the task domain (something that Cirrus has none of) in order to interpret the whole protocol, both actions and

non-action commentaries. Cirrus then tests the adequacy of the human's analysis and deepens it by trying to find selection strategies that will accurately postdict the subject's actions. If some sections of the action protocol are not adequately fit, then the human theorist can re-examine the protocol at those points, revise the problem space, and run Cirrus again. In short, Cirrus is a data analysis tool and not an automated protocol analyst.

This paper describes the current version of Cirrus. (An earlier version is reported in VanLehn and Garlick, 1987.) The first section describes the theory of problem solving that Cirrus assumes. Although the theory is quite standard and noncommittal, it is nonetheless the source of most of Cirrus' limitations as an analytical tool. The second section describes how Cirrus works. The third section discusses Cirrus' analysis of protocols from two different task domains. The last section discusses our plans for removing the current restrictions imposed by Cirrus' theory of problem solving.

ASSUMPTIONS ABOUT PROBLEM SOLVING

This section presents some assumptions about the representation and interpretation of procedural knowledge. All of the assumptions are fairly standard in the field, but the particular combination of assumptions used in Cirrus need to be made explicit. First, the basic problem solving architecture will be described, then the assumptions behind it will be discussed.

The basic cycle of problem solving is to (1) select a goal, (2) select an operator that is relevant to achieving that goal, (3) execute the operator, and (4) delete the goal. Some operators are primitive, in that executing them changes the state of the problem. Other operators are macro-operators, in that executing them causes new goals to be created. This basic cycle is initiated with a top level goal; it finishes when there are no goals left.

The knowledge representation consists of (1) a set of operators, (2) a strategy that determines which operator to choose for any given goal, (3) a strategy for determining which goal to choose, and (4) a state description vocabulary, which is a list of the attributes of states that are considered relevant for making strategic decisions. The user provides Cirrus with the set of operators, the operator selection strategy and the state description vocabulary. Cirrus infers the subject's goal selection strategy. Soon, Cirrus will also be able to infer the subject's operator selection strategy as well. This capability is a standard technique in machine learning (see, e.g., Langley and Ohlsson, 1984), and we anticipate no problems incorporating it in Cirrus. Thus, Cirrus will require only the problem space (i.e., operators and state description language) from the user, and will infer the rest of the subject model itself.

There are several tacit features of this problem solving architecture that distinguish it from others in the GPS/STRIPS class. First, it lacks GPS's commitment to difference reduction as the method for choosing operators. The operator selection strategy can, in principle, be any function of the current state, and not one that reduces the difference between the current state and the goal. Currently, Cirrus uses a simple representation for operator selection strategies. Each operator has a condition attached to it. After the architecture has found which operators are relevant to the current goal, it tests the conditions attached to the operators. If exactly one operator has a true condition, it is selected; otherwise an impasse occurs (Brown & VanLehn, 1980). Currently, Cirrus does not model subject's reactions to impasses.

In order to back up during problem solving, most GPS-style problem solvers have a state selection

phase that immediately precedes the goal selection phase. Cirrus does not, in part because backing up is not that common, and when it does occur, the subject often just starts the problem over (Newell & Simon, 1972). When starting over occurs, it is usually easy for a human analyst to spot, so the coder can merely given Cirrus a protocol that has two solutions of the same problem, the first of which happens to end in failure. Thus, Cirrus can "handle" the most common types of backup, albeit crudely.

Like GPS, Cirrus does not model learning during the course of problem solving.

Like GPS, Cirrus has very limited planning abilities. Subjects sometimes plan ahead by mentally simulating the next few actions. Cirrus can not model this. Nor can Cirrus model abstraction planning, such as that used by Newell and Simon's logic subjects. Nor can Cirrus model envisioning, wherein a subject mentally simulates the working of a physcial device. Cirrus can not do these things because it only represents one problem state. Usually, the current problem state represents current state of the real, physical world, although it can represent an imaginary world, as when the subject is doing mental mulitplication. Cirrus can not represent someone who is working in both a mental world (often as a way of developing a plan) and a real world. This means that the only kind of planning that Cirrus can represent is operator subgoaling, wherein one develops a stack of subgoals before finally arriving at an operator that can be executed directly (e.g., "In order to get from Pittsburgh to Montreal, I'll fly; but a precondition of flying is being at the airport, so I'll take a cab to the airport; but a precondition of taking a cab is ...").

However, Cirrus does have a flexible approach to subgoaling. It does not insist on a depth-first traversal of the goal hierarchy. Instead, it assumes people have a goal selection strategy. For instance, if taking a cab spawns the subgoals of (1) being at a cab and (2) having enough money to take a cab, then the goal selection strategy would have to decide which of these new goals to work on. Indeed, it could even decide to work on an older goal, such as obtaining money for airfare. Thus, Cirrus can model subjects who jump around in the goal hierarchy. This has turned out, somewhat surprisingly, to be necessary even for modeling simple skills, such as subtraction (VanLehn & Ball, 1987).

The goal selection strategy of Cirrus is represented as a set of conditional preferences of the form "Prefer <goal-1> over <goal-2> when <condition>." For instance, in subtraction, it is important to execute borrowing actions that effect the top digit of a column before one answers the column. To represent this, one can use preferences such as "Prefer (Add10 ?C1) over (Diff ?C2) when C1=C2." This says that when both the goal of adding ten to the top of a column and the goal of taking the difference in a column occur, and they refer to the same column, then one should do the addition of ten first. Such preferences are used in many contemporary problem solvers, such as Soar (Laird, Newell, & Rosenbloom, 1987) and Prodigy (Minton et al., 1987).

Cirrus has been described as problem solver. However, that is only a small fraction of its job. Its main task is to find the goal selection strategy of the subject (and, soon, the operator selection strategies as well). When those are found, the complete model of the subject is executed by the problem solver in order to measure its fit to the protocol.

THE ANALYSIS METHOD

This section describes the techniques used by Cirrus to infer the goal selection strategy. Note that, in contrast to the preceding section, these processes are *not* claimed to be psychological processes. Neither the subject nor the human analyst does anything remotely similar to what is described in this

section.

The first step in the analysis of an action protocol is to parse the protocol using the operator set provided by the user. The algorithm used is a modification of a standard top-down algorithm for parsing with context-free grammars. The result, in the ideal case, is a single tree for each problem. The tree represents the goal-subgoal decomposition that the subject took on that problem. The leaves of the tree are the primitive operator applications that constitute the subject's actions. In the less-than-ideal case, parsing a problem produces more than one tree. Cirrus asks the user to choose. In the worst case, some problems can not be parsed at all. This indicates an inadequacy in the set of operators which the user must correct.

The second step in the analysis is to convert each goal-subgoal tree into the trace that the problem solver would produce if it had generated this tree. The trace contains each operator selection event and each goal selection event that the problem solver would have had to perform. Cirrus generates this trace by traversing the tree, and outputing the appropriate selection events each time it visits a goal in the tree. Unfortunately, the trace is not always uniquely determined. Sometimes the tree can be traversed in hundreds of ways while still remaining faithful to the chronology of the action protocol. Cirrus currently uses heuristics to guide the tree traversal. The heuristics are based on the assumption that solvers tend to minimize their working memory load. The heuristics guide the tree traversal along a path that minimizes the number of active goals. It is not yet clear whether this heuristic will always be a good one or how to detect when it has led the analysis off on a garden path. This would be a good place for Cirrus to have access to the non-action parts of the verbal protocol, because some of the subject's comments might indicate in which order the goals are considered.

The next step in the analysis is to convert the goal selection events into preferences. This occurs in two phases. First, preferences are constructed without conditions attached to them. Such preferences can be either *consistent* with a goal selection event, or *inconsistent*, or *irrelevant*. Thus, if the preference is "Prefer A over B", and the event is that goal A is chosen when both A and B exist, then the preference is consistent with the event. The preference would be inconsistent with the event if B were chosen instead of A. The preference is relevant only to events where both A and B exist. Preferences are constructed only if they are consistent with at least one event.

The second phase builds conditions for preferences. If a preference is consistent with some events and inconsistent with others, then a condition is induced that is true of all the consistent events and false of all the inconsistent ones. The condition is a boolean combination of attribute-value pairs drawn from the state description vocabulary. (This is the main reason for having a state description vocubulary.) The ID3 algorithm is used to perform condition induction (Quinlan, 1983). There are some biases built into ID3 as well as all other condition induction algorithms. We have experimented with several other induction algorithms: the Chi-squared variant of ID3 (Quinlan, 1986), the standard version space algorithm (Mitchell, 1982), a noise-handling version space algorithm (Mitchell, 1978), and some algorithms of our own. Our experience is that the bias inherent in the condition inducing algorithm is not as important as the bias inherent in the state description vocabulary. An inadequate vocabulary will hurt them all, and a good vocabulary allows all of them to succeed, more or less. We use ID3 because it seems best at handling noisy data.

The last step in the analysis is to test the model by executing it and comparing its "protocol" with

the subject's. This step is purely for the user's benefit, because the current version of Cirrus can not go back and reconsider the heuristically determined choices it made during analysis. The user is responsible for feeding the results of the analysis back into the Cirrus, typically via changes in the state description vocabulary.

The original version of Cirrus (VanLehn & Garlick, 1987) used a completely different representation for goal selection stategies and a completely different method for inferring it. Kowalski and VanLehn (1988) discuss its shortcomings and the experiments that led to the current version of Cirrus.

RESULTS

Cirrus has been extensively tested on protocols collected from eight grade-school students solving multi-column subtraction problems (Kowalski & VanLehn, 1988). Preliminary testing has also been done on two protocols collected via the use of Sketch, a tutoring system that trains students to graph transcendental equations (Trowbridge, Larkin & Sheftic, 1987). This section discusses our experiences using Cirrus to analyze these data.

The subtraction protocols were collected as part of a study reported earlier (VanLehn & Ball, 1987), so the subjects, methods and results will not be detailed here. The chief finding is that eight subjects in a biased sample of 26 third-graders executed subtraction procedures in non-standard orders. For instance, some students would do all the borrowing in a problem first, moving from right to left, then fill in the answers to the columns, moving from left to right. A second finding is that students frequently change their execution ordering, often in the middle of solving a problem. This makes it seem likely that they are using a non-standard goal selection strategy that is conditional on the characteristics of the problem state, rather than having learned a non-standard algorithm for solving subtraction problems. However, when these protocols were analyzed by hand, we were only partially successful at finding goal selection strategies that would explain the subjects behavior. Up to one-third of the choice points in some protocols were left undetermined (VanLehn & Ball, 1987). The tedium of trying to improve this fit was a major motivation for the Cirrus project.

When Cirrus analyzes the action protocols of the eight non-standard order subjects, the subject models obtained are nearly perfect. Except for one ambiguous choice point in the protocol of one subject, the models infered by Cirrus exactly match the protocols. Not only does Cirrus do the job it was designed to do, it outperforms human analysts in its ability to match the students' protocols.

However, perfect matching of protocols is to be expected from Cirrus. The induction techniques used in Cirrus are powerful enough to analyze any goal selection strategy, provided that the user has the patience to keep augmenting the state description vocabulary until Cirrus finds appropriate conditions for all the preferences. Thus, the appropriate measure of Cirrus' performance is whether the preferences it finds make sense as goal selection strategies. This is not an easy evaluation to make objectively, so we can only present our experiences. In general, when Cirrus builds a small condition (3 to 5 attributes), our intuition agrees with its condition; the student really does seem to have that preference. However, when Cirrus builds a large, deeply nested condition (7 to 9 predicates), our intuition is that the subject probably has no preference (i.e., they choose whimsically between these two types of goals) or has a define preference but occasionally slips in applying it. However, during the early stages of using Cirrus on these protocols, most of the large, uninterpretable conditions turned out to be due to inadequacies in the state

description vocabulary.

The Sketch data were collected directly from the user interface of a tutoring system. Preliminary analysis of the Sketch protocols showed that Cirrus is a feasible system to use to analyze these in greater depth. The protocols share a characteristic with the subtraction protocols in that subjects tend to alter subtask ordering depending on the problem context. Cirrus has done an adequate job of characterizing the problem features that lead to the reordering of subtasks, although a theorist has not yet spent time in the loop with Cirrus refining the state description vocabulary to fully capture the patterns in the data.

DISCUSSION AND FURTHER WORK

Cirrus has potential both as a data analysis tool for scientists and as the student modelling component of an intelligent tutoring system (VanLehn, 1988). Indeed, Cirrus grew out earlier two student modellers for the subtraction task, Debuggy (Burton, 1982) and ACM (Langley & Ohlsson, 1984). Like Cirrus, these systems analyze data off-line, because they are too slow for real-time use in an intelligent tutoring system. A more important limitation is that all three systems, Cirrus, ACM and Debuggy (but not IDebuggy, a variant of Debuggy), expect data from all the problems to be available initially, whereas real-time student modelling requires taking in problem solutions as they are generated, and revising the student model incrementally. Incremental versions of all the analysis algorithms used by Cirrus exist in the machine-learning literature, but we have not yet tried to convert them because we are, at present, more interested in developing Cirrus as a tool for scientists.

The most important direction for improving Cirrus is to relax the assumptions about problem solving that are built into it. We would like to use Cirrus to analyze protocols of college students solving physics problems, which were graciously given to us by M.T.H. Chi. Almost all the work on physics to date has used coarse-grained protocol analyses, such as constrasting the order in which novices and experts write equations. This misses some interesting behavior, such as planning and envisioning, which is revealed in some of the subjects' verbal commentary. As mentioned earlier, Cirrus' assumptions do not allow for "dual world" problem solving, so this kind of behavior can not be modelled. Significant extension to Cirrus will be required.

The current version of Cirrus is definitely useful, although it is still a research prototype rather than robust, distributable software. Nonetheless, it demonstrates that human protocol analysis can be aided by computer-based tools. It has taken longer to develop these tools than it took to develop statistical packages because the underlying machine learning technology was developed only recently. Nonetheless, the way seems clear for further development of protocol analysis tools, which should allow much easier interpretation of a particularly rich and revealing window on human cognition.

REFERENCES

Bhaskar, R. & Simon, H. A. (1977). Problem solving in a semantically rich domains: An example from engineering thermodynamics. *Cognitive Science*, *1*, 193-215.

Brown, J. S. & VanLehn, K. (1980). Repair Theory: A generative theory of bugs in procedural skills. *Cognitive Science*, *4*, 379-426.

Burton, R. B. (1982). Diagnosing bugs in a simple procedural skill. In D. H. Sleeman & J. S. Brown (Eds.), *Intelligent Tutoring Systems*. New York: Academic. 157-183.

Fisher, C. (1988). Advancing the study of programming with computer-aided protocol analysis. In G. Olson, E. Soloway & S. Sheppard (Ed.), *Empirical Studies of Programmers*. Norwood, NJ: Ablex.

Kowalski, B. & VanLehn, K. (1988). *Induction of partial orders beats classification: Improvements to the Cirrus protocol analysis system* (Tech. Rep. PCG-15). Dept. of Psychology, Carnegie-Mellon University.

Laird, J. E., Newell, A., and Rosenbloom, P. S. (1987). Soar: An Architecture for General Intelligence. *Artificial Intelligence, 33*, 1-64.

Langley, P. & Ohlsson, S. (1984). Automated cognitive modeling. In *Proceedings of AAAI-84*. Los Altos, CA.: Morgan Kaufman,

Minton, S., Carbonell, J.G., Etzioni, O., Knoblock, C. & Kuokka, D.R. (1987). Acquiring effective search control rules: Explanation-based learning in the Prodigy sytem. In P. Langley (Ed.), *Proceedings of the Fourth International Workshop on Machine Learning*. Los Altos, CA: Morgan Kaufman.

Mitchell, T.M. (1978). *Version spaces: An approach to concept learning* (Tech. Rep. STAN-CS-78-711). Computer Science Department, Stanford University.

Mitchell, T.M. (1982). Generalization as search. *Artificial Intelligence, 18*, 203-226.

Newell, A. & Simon, H. A. (1972). *Human Problem Solving*. Englewood Cliffs, NJ: Prentice-Hall.

Quinlan, J. R. (1983). Inductive inference as a tool for the construction of high-performance programs. In R. S. Michalski, T. M. Mitchell & J. Carbonell (Eds.), *Machine Learning*. Palo Alto, CA: Tioga.

Quinlan, J. R. (1986). The effect of noise on concept learning. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Ed.), *Machine Learning: An Artificial Intelligence Approach. Volume II*. Los Altos, CA: Morgan Kaufman.

Trowbridge, D., Larkin, J. & Scheftic, C. (1987). A computer-based tutor forgraphing equations. In *Proceedings of National Education Computer Conference*. Eugene, OR: ICCE,

VanLehn, K. (1988). Student modelling. In M.C. Polson & J.J. Richardson (Ed.), *Foundations of Intelligent Tutoring Systems*. Hillsdale, NJ: Erlbaum.

VanLehn, K. & Ball, W. (1987). *Flexible execution of cognitive procedures* (Technical Report PCG-5). Carnegie-Mellon University, Dept. of Psychology.

VanLehn, K. & Garlick, S. (1987). Cirrus: an automated protocol analysis tool. In Langley, P. (Ed.), *Proceedings of the fourth Machine Learning Workshop*. Los Altos, CA: Morgan-Kaufman.

Waterman, D.A. & Newell, A. (1971). Protocol analysis as a task for artificial intelligence. *Artificial Intelligence, 2*, 285-318.

Waterman, D.A. & Newell, A. (1973). *PAS-II: An interactive task-free version of an automatic protocol analysis system* (Tech. Rep.). Department of Computer Science, Carnegie-Mellon University.

THREE KINDS OF CONCEPTS?

Barbara C. Malt

Department of Psychology
Lehigh University
Bethlehem, Pennsylvania

For many years researchers assumed that all concepts could be adequately described by a set of defining (necessary and sufficient) features: analyses of concepts such as bachelor into features such as "unmarried" and "male" were thought to be extendible to ordinary concepts such as bird and chair (Katz & Fodor, 1963). A more recent view is that features are distributed across members of most categories in a "family resemblance" fashion such that features will be true of some subset of members of the category, but will never be singly necessary or jointly sufficient for category membership (Rosch & Mervis, 1975; Wittgenstein, 1953). Under this view, many common concepts consist of a set of features that are only associated with the category with some probability.

The importance of non-defining features in concept representations is by now undeniable (Rosch & Mervis, 1975; Smith, Shoben, & Rips, 1974). Their importance does not, however, preclude the possibility that defining features are also involved in concept representations. Armstrong, Gleitman, and Gleitman (1983) have shown that patterns of reaction times and typicality ratings taken as support for the family resemblance view can also be obtained for concepts such as bachelor that presumably do involve defining features. Others (e.g. Osherson & Smith, 1981; Rey, 1985) have argued on logical grounds that a pure family resemblance view may not be sufficient to account for facts about how we use concepts.

To solve these problems, concepts are now sometimes said to contain both non-defining features and a concept "core" of defining features (e.g. Armstrong et al., 1983;, Keil, 1986; Medin & Smith, 1984; Rey, 1985). Yet even this sort of hybrid proposal is not entirely satisfactory. A variety of evidence suggests that artifact concepts (toy, furniture, game, etc.) may differ in important ways from natural kind concepts (bird, fish, tree, etc.) For instance, even the most atypical members of natural kind categories seem to truly belong to their category (e.g. a penguin is a full-fledged member of the category "bird" no matter how little it resembles other birds), whereas atypical members of artifact categories seem to only "sort of" belong to the category (e.g. a lamp is a very marginal member of the category "furniture") [Rey, 1985; Lakoff, 1987]. Similarly, unclear cases seem to arise for natural kind categories due to insufficient knowledge about critical properties of the categories (e.g. in trying to decide whether a tomato is a fruit or a vegetable), while they seem to arise for artifact categories due to lack of clear boundaries between the categories (e.g. in trying to decide whether a TV is furniture or an electrical appliance) [Rey, 1983; Malt, 1985]. These observations suggest that natural kind concepts possess something much more core-like than artifact concepts do.

But if, as previous investigations (e.g. Rosch & Mervis, 1975) suggested, people often do not know necessary and sufficient features for natural kinds,

then what underlies the difference between the two types of concepts? First, the nature of the feature information may still differ between the two in a subtle but important way: While natural kind concepts may include at least vague notions about the existence of core features, artifact concepts may not include anything resembling potential core features at all. Second, very different beliefs may be held about the completeness of the features as a description of the category they represent: While natural kind concepts may include a belief that more complete core information is in principle knowable, and is in fact known by experts, artifact concepts may include a belief that the characteristic features represented are all there is to know about the category.

The experiments reported below explore the possibility that these two separate types of concepts exist. They also investigate the possibility that a third kind of concept exists: those such as bachelor and grandmother for which the traditional analysis in terms of defining features have seemed most satisfactory (henceforth to be referred to as "relational kinds", since their definition often involves a relationship between one person or object and another). These latter concepts would presumbably include both necessary and sufficient features and a belief that knowledge of relevant features is complete.

EXPERIMENT 1

Lakoff (1972) argued that linguistic hedges (e.g. "loosely speaking;" "technically") differ among themselves in the kinds of features that they refer to. If so, looking at acceptability judgments for sentences combining different hedges with various category terms should shed light on what sorts of features the concept representations contain. Subjects in this experiment read sentences containing a hedge and a category term (e.g. "Loosely speaking, that's a bird" and "Technically, that's a piece of clothing"), and judged whether each sentence was sensible or not.

The four hedges used were by "by definition," "technically," "according to experts," and "loosely speaking." A variety of arguments (Lakoff, 1972; Malt, 1985) suggest that "by definition" should only be acceptable in combination with categories for which one knows defining features; "technically" should be acceptable with those for which defining features are either known or are believed to exist; "according to experts" should be acceptable only when the complete meaning of the word is assumed to be known to experts in a domain but not to everyone else; and "loosely speaking" should be acceptable when a category does not have clear boundaries. Thus if the proposed distinctions among concepts are correct, "by definition" should be most acceptable with relational kind categories such as "bachelor"; "technically" with both relational kinds and natural kinds; "according to experts" with natural kinds; and "loosely speaking" with artifact categories.

A previous experiment (Malt, 1985) tested similar predictions using a total of twelve categories. Results supported the predictions: for instance, sentences combining "loosely speaking" with natural kind terms (e.g. "Loosely speaking, that's a tree") received low ratings, while sentences combining "loosely speaking" with artifact terms (e.g. "Loosely speaking, that's a tool") received significantly higher ratings. People clearly treated the different concepts differently in this task, and the differences corresponded

to the proposed distinctions in the nature of the representations. The present experiment was designed to provide additional support for these results in two ways: first, to establish their generality by using 12 new categories; and second, to address a possible criticism of the earlier experiment. In the previous experiment, the hedge used to differentiate natural kind concepts from the others was "scientifically speaking." "Scientifically speaking" may have been rated as most acceptable with natural kind terms simply because they are the only categories studied by people called scientists, not because part of their meaning is known only to experts. This hedge was therefore replaced by the hedge "according to experts." Subjects should find "according to experts" acceptable only if they believe the term in question has some aspect of its meaning that is known to an expert but not to the average language user.

Method

Twenty-four Lehigh University undergraduates participated. Four artifact terms (sport, toy, appliance, machine), four natural kind terms (star, planet, mineral, grass), and four relational kind terms (orphan, majority, hole-in-one, stealing) were used. As noted above, the four hedges were "technically," "by definition," "loosely speaking," and "according to experts." Each hedge was paired with each category, and every subject saw all target stimuli. Target stimuli were mixed with filler sentences involving hedges and categories not of interest in this experiment, and two random orders of sentences were constructed. The rating scale was from "1" to "7," with high numbers indicating high judged sensibility and low numbers indicating low judged sensibility.

Results

Mean ratings for each category type and hedge combination are given in the table below.

	relational kind	natural kind	artifact
by definition	5.8	5.5	5.3
according to experts	3.4	5.5	3.7
loosely speaking	3.3	3.0	4.2
technically	5.5	5.1	5.3

TABLE 1

Results replicated those of Malt (1985): As predicted, "loosely speaking" was judged more sensible with artifact categories than with the others; "according to experts" was judged most sensible with natural kind categories, and "by definition" was judged most sensible with relational kinds. The pattern for "technically" was not as predicted; it received approximately equal ratings for all category types. This finding is consistent with a marginal result for "technically" in the earlier experiment,

and will be discussed later. For the other three hedges, ratings again support the proposal that artifact, natural kind, and relation kind concepts differ in the information contained in the mental representation. (Note that column comparisons are not meaningful, since the hedges themselves vary somewhat in familiarity and "goodness").

Statistical analysis confirms these observations. An overall ANOVA showed a significant interaction between hedge and category type, $F(6, 138) = 33.54$, $p < .001$. Individual contrasts on rows showed that "loosely speaking" was judged more sensible with artifact categories than with the others [$F(1, 69) = 22.05$, $p < .001$]; "according to experts" was judged most sensible with natural kind categories [$F(1, 69) = 61.25$, $p < .001$]; and "by definition" was judged most sensible with relational kinds [$F(1, 69) = 3.20$, $.10 < p < .05$]. Ratings for "technically" did not conform to the predicted pattern [$F(1, 69) = 0$.]

EXPERIMENT 2

The second experiment asked about beliefs about concept completeness more explictly. Subjects were told to imagine they were trying to teach object names to a visitor from another planet. They were then given scenarios in which they encounter an artifact, natural kind, or relational kind exemplar that is difficult to classify. For instance, a scenario with natural kinds was as follows:

> "In an orchard on the outskirst of town, you see a tree that seems to you to be sort of halfway between an orange tree and a lemon tree. You explain the dilemma to the visitor, and you say:
>
> a. "If I could think about it long enough, I could tell you which it is."
> b. "We'd have to ask an expert to tell us which it is."
> c. "Well, I guess you can call it whichever you want."

If people actually know the defining features for a category, they should believe that they will be able to make a correct classification decision even in difficult cases, and they should choose option (a). If they believe that they don't have all the relevant information but an expert does, they should feel that consulting someone else would be necessary [option (b)]. If they believe that they have all the possible information but the boundaries of the categories are fuzzy, they should feel that the classification is simply indeterminate [option (c)].

Method

Forty Lehigh University undergraduates participated. There were 22 target scenarios, consisting of 8 natural kind scenarios, 8 artifact scenarios, and 6 relational kind scenarios. Each scenario involved a description of an object that appeared to be halfway between two familiar categories. The pairs of categories for natural kinds were robin-sparrow; oak-maple; sardine-anchovy; rose-carnation; chicken-turkey; orange tree-lemon tree; trout-bass, and marigold-dandelion. The pairs for artifacts were chair-couch; boat-ship; shirt-blouse; hide and seek-tag; bookcase-shelf; car-truck; socks-stockings. The pairs for relational kinds were bachelor-widower;

triangle-square; prime number-odd number; homerun-triple; grandmother-aunt; and even number-odd number. These pairs were obtained in a preliminary experiment in which a separate group of 24 subjects was given the first member of each pair and standard instructions for obtaining linguistic contrast sets ("If it's not an X, what might it be?") The most frequently listed response was used as the second member of each pair.

Target scenarios were mixed with filler scenarios involving categories and response options not of interest in this experiment. Two different stimulus sets were constructed, each containing half the scenarios of each category type. Half the subjects received each set. Two versions of each set were used, differing in the order of scenarios and the order of response options.

Results

The table below gives the number of subjects out of 40 who chose each option on the majority of trials for each of the three category types.

	option (a) "..think.."	option (b) "..expert.."	option (c) "..whichever.."	mixed
artifact	1	5	25	9
natural kind	1	30	5	4
relational kind	7	10	21	2

TABLE 2

Chi-square values were computed comparing response patterns for each category type to the pattern expected if choices were random. Responses for all three category types diverged significantly from randomness, $X^2 = 32.32$ for artifacts and $X^2 = 41.18$ for natural kinds, $p < .001$ for both; $X^2 = 8.58$, $p < .025$ for relational kinds.

The pattern of responding clearly supported the predictions for artifact and natural kind categories. Subjects tended to choose option (c) for artifacts and option (b) for natural kinds, suggesting that they believe classificiation is not clear-cut for artifacts but is for natural kinds and can be determined by experts. Response patterns for both types diverged significantly from randomness.

Contrary to prediction, people tended to choose either option (b) or option (c) for relational kinds. One interpretation is that relational kinds are not qualitatively different from artifacts or natural kinds. An alternative, however, is that subjects found it odd to even propose the existence of an entity halfway between two relational kind categories and assumed that their knowledge would be insufficient to deal with such anomalous cases. This possibility would be further evidence for the well-defined nature of these categories and will be explored in future experiments.

DISCUSSION

Armstrong et al.'s (1983) experiments mentioned earlier demonstrate that standard reaction time and typicality rating tasks cannot provide definitive evidence about the existence of defining features in a concept. Alternative tasks must be found that reflect underlying knowledge about a category rather than performance strategies, and that do not require conscious access to the knowledge. The experiments reported here demonstrate that people make distinctions among concepts in tasks that call on their linguistic intuitions rather than performance in a speeded task. Thus, the linguistic judgment tasks used in these experiments appear to provide an approach that is sensitive to subtle variations in the content of concepts.

The most striking outcome of the two experiments is that subjects show a sharp distinction in the extent to which they feel natural kind and artifact concepts can be used in a loose, non-technical way. In both Experiment 1 and the earlier hedges study, subjects found "loosely speaking" much more acceptable with artifacts than with natural kind terms, while they found "according to experts" (or "scientifically speaking" in the earlier study) more acceptable with natural kinds. A subsequent experiment has confirmed that the result holds even when the natural kind and artifact categories are closely equated for level of abstraction; thus, this result is not an artifact of more concrete concepts in the natural kind domain. In Experiment 2, subjects consistently chose the response "You can call it whichever you want" for ambiguous artifacts, while they chose "We'd have to ask an expert" for natural kinds. These results together clearly indicate that the average college student participant believes there is a component of meaning to natural kind terms that may not be present in his or her own mental representation of the category. A similar belief does not seem to exist for artifacts; the students appear to be willing to use the terms in a looser fashion and to believe that such as use is appropriate. These results thus support the idea that natural kind and artifact concept representations differ, and further that beliefs about completeness of knowledge are an important component of a concept in addition to the actual features that are represented.

Given that the linguistic judgments are sensitive to variations in concept representations, one question that remains is why the hedge "technically" did not produce the predicted difference among the category types in Experiment 1 and produced only a marginal difference in the earlier hedges study. Subjects' comments provide some insight on this issue. Several pointed out that it is often possible to imagine contexts where it would be appropriate to speak technically of artifact categories. For instance, a department store might establish a rule to determine when an object should be displayed as sports equipment vs. when it belongs in the toy department. Thus a technical definition can potentially exist for an artifact term even if the general usage is not of this nature.

Finally, it important to note that although there was enough consistency among the natural kind, artifact, and relational kind terms to produce significant differences among the three groups, there was also noticeable item variability in the experiments reported here and in the earlier hedges study and the control study mentioned above. For example, "grass" received received

relatively low ratings for "according to experts" compared to the other natural kind terms, and "ship" received a relatively high number of choices of "you'd have to ask an expert" compared to other artifact categories.
In almost all cases, the discrepancies seem to involve highly familiar natural kinds being treated more like artifact categories and vice versa. These observations suggest that category type per se is not the only determiner of concept representation. Familiar natural kind concepts may tend to be represented more like artifact categories are, and concepts for unfamiliar artifacts used mainly in restricted settings may tend to resemble those for natural kinds. This possibility is being explored in studies in progress.

REFERENCES

Armstrong, S. L., Gleitman, L. R., & Gleitman, H. (1983). What some concepts might not be. Cognition, 13, 263-308.

Katz, J. J. & Fodor, J. A. (1963). The structure of a semantic theory. Language, 39, 190-210.

Keil. F. C. (1986). The acquisition of natural kind and artifact terms. In W. Demopoulos & A. Marras (Eds.), Language Learning and Concept Acquisition: Foundational Issues. Norwood, NJ: Ablex Publishing Corporation.

Lakoff, G. (1972). Hedges: A study in meaning criteria and the logic of fuzzy concepts. In Papers from the Eighth Regional Meeting, Chicago Linguistics Society. Chicago: University of Chicago Linguistics Department.

Lakoff, G. (1987). Women, Fire, and Dangerous Things: What Categories Reveal about the Mind. Chicago: The University of Chicago Press.

Malt, B. C. (1985). Hedges and the mental representation of categories. CC-AI: The Journal for the Integrated Study of Artificial Intelligence, Cognitive Science, and Applied Epistemology, 2, 13-23.

Medin, D. L. & Smith, E. E. 1984. Concepts and concept formation. Annual Review of Psychology, 35, 113-138.

Osherson, D. N. & Smith, E. E. (1981). On the adequacy of prototype theory as a theory of concepts. Cognition, 9, 35-58.

Rey, G. (1985). Concepts and conceptions: A reply to Smith, Medin, and Rips. Cognition, 19, 297-303.

Rosch, E. & Mervis, C.B. (1975). Family resemblances: Studies in the internal structure of categories. Cognitive Psychology, 7, 573-605.

Smith, E. E., Shoben, E. J., & Rips, L. J. (1974). Structure and process in semantic memory: A featural model for semantic decisions. Psychological Review, 81, 214-241.

Wittgenstein, L. (1953). Philosophical Investigations. New York: MacMillan.

Constructing Coherent Text using Rhetorical Relations*

Johanna D. Moore
UCLA Department of Computer Science
and
USC/Information Sciences Institute

Cecile L. Paris
USC/Information Sciences Institute

Introduction

Language is planned behavior: speakers produce complex combinations of utterances to cause certain effects on their hearers, such as persuading them, informing them or requesting something from them [Aus75], [Gri75], [CP79], [AP80], [App81], [CL85]. Researchers in the field of text generation have concentrated their efforts on producing a computational model of the process of generating coherent text to achieve a speaker's conversational goals. But producing text is only a small part of a much broader cognitive faculty of language which includes such tasks as responding to follow-up questions, elaborating or clarifying texts that were not completely understood by the hearer, and learning rhetorical strategies to achieve conversational goals. To perform these other tasks, it is not sufficient to simply produce coherent text. An agent must understand the text it produces in terms of how each part of the text relates to the others and how the complete text achieves its goal.

Computational models of text generation put forth to date fall into two extremes: *plan*-based approaches and *schema*-based approaches. Using a plan-based approach [CP79] [App81], a text plan is produced by reasoning about the beliefs of the hearer and speaker and the effects of various speech acts (e.g., REQUEST, INFORM) on these beliefs. This approach thus captures the intended effects of an utterance on the hearer. However, it does not include rhetorical knowledge which is knowledge about how to combine individual speech acts into larger pieces of coherent text. As a result, systems using the plan-based approach cannot produce texts to describe objects, compare and contrast two entities, explain how a device works, or justify a result.

To produce larger bodies of text, researchers have turned to schema-based approaches (e.g., [McK82], [McC85], [Par87]) which employ script-like structures (schemata) to generate coherent multi-sentential texts achieving a given discourse goal. Schemata encode standard patterns of discourse using rhetorical predicates, but do not include knowledge of how the various parts of the schema relate to one another or what their intended effect on the hearer is. As a result, if the hearer does not understand the text, it is very hard to recover, i.e., to know which part of the schema failed to achieve its effect on the hearer and produce text to correct the misunderstanding.

Recently, Hovy has begun to incorporate rhetorical relations into the plan-based approach to produce multi-sentential texts, but has concentrated solely on ordering a set of propositions [Hov88]. His system assumes that the information to be said is chosen in advance and given to the system as input. We believe that the tasks of choosing what to say and how to organize it cannot be divided in this way: they are intertwined and can influence one another. The choice of what to say next is in fact largely dependent on what has already been said.

In this work, we are concerned with generating coherent multi-sentential texts, deciding both what to say and how to organize it, in such a way that a system can recover from failure. We want our generation system to explicitly record how the different parts of the text are related and how they affect the hearer's beliefs. To this end, we propose a plan structure which relates discourse goals, expressed in terms of their intended effects on the hearer, to the rhetorical means used to achieve them. Rhetorical means might be low

*The research described in this paper was supported under DARPA Grant #MDA 903-81-C-0335. The authors would like to thank Eduard Hovy and William Swartout for their comments and suggestions on this paper.

level speech acts, such as INFORM, or rhetorical relations, as defined in Rhetorical Structure Theory (RST) [MT87], such as MOTIVATION.

This paper presents the structure of the text planning operators in our model and the planning mechanism that uses these operators to construct text. To illustrate how a text plan is produced, we show an example. We discuss the advantages of this approach and show how it can be seen as a computational model that unifies both the schema-based and the plan-based approaches presented to date.

The Plan Language and the Planning Mechanism

The planning mechanism we use is a conventional top-down hierarchical expansion planner [Sac75]. The planner begins with a high-level discourse goal of the speaker and successively refines it into other discourse subgoals or rhetorical means for achieving them. This process continues until the entire plan is refined into primitive operators, i.e., speech acts. Discourse goals are represented in terms of the effects that the speaker intends his or her utterance to have on the hearer's knowledge. Following Hovy in [Hov88] we have adopted the terminology for expressing beliefs developed by Cohen and Levesque in their theory of rational interaction [CL85].[1]

There are two types of plan operators in our system: operators whose effects are characterizations of the hearer's beliefs, and operators that achieve a particular rhetorical relation. Figure 1 shows a plan operator whose effect is the state where the speaker believes that the hearer and speaker mutually believe that it is a goal of the hearer that a given act eventually be done by the hearer. Figure 2 shows a text plan capable of achieving the rhetorical relation MOTIVATION which relates an act to the goal it achieves. When instantiated, the plan will inform the hearer that the act is a step towards achieving the goal and may elaborate as to why this is the case.

Each plan operator consists of:

- **an effect:** a characterization of what goal(s) this operator can be used to achieve.
- **a constraint list:** a list of conditions that must be true before the operator can be applied. Constraints may refer to facts or relations in the domain of discourse or states of the hearer's knowledge. Some of the constraints correspond to the constraints on a nucleus and satellite as specified in RST. Additional constraints had to be introduced in order to actually *construct* text, as RST was developed as a tool for *analyzing* text.
- **a nucleus and a satellite:** each is a partially ordered sequence of operators that, when executed, may achieve the effects of this operator. The nucleus is the focus of the text produced by this operator and must be present. It could be a speech act or a desired state, which can later be expanded. Satellites are additional parts of the plan. They may be required or optional.

When a goal is posted, the planner finds all the plan operators whose effects match the goal and whose constraints are satisfied. It selects one such operator and posts the steps listed in the nucleus and satellite as subgoals to be satisfied. The completed text plan is given as input to a text generation system, Penman in our case [MM83].

An Example

One of the prototype systems we are building is an explanation component for an expert system that aids users in improving their LISP code by recommending transformations that enhance the user's code along the dimensions of readability, maintainability, or efficiency.[2] The most natural way for a dialogue to begin is with a recommendation by the system. The user is then free to ask questions about this recommendation.

Suppose for example that the user wanted to enhance the readability of the program under consideration. The expert system might wish to recommend that the user replace calls to the function CAR with calls to the function FIRST. To do so, the expert system would post the following discourse goal to the text planner:

```
(BMB S H (GOAL H Eventually(DONE H replace-1)))
```

[1]Space limitations prohibit an exposition of their terminology in this paper. For clarity, we will paraphrase the terminology in English in our examples.

[2]Details regarding the expert system can be found in [NSM85].

EFFECT: (BMB S H (GOAL H Ev(DONE H ?act)))
CONSTRAINTS: none
NUCLEUS: (BMB S H (GOAL S (GOAL H Ev(DONE H ?act)))
SATELLITES: (BMB S H (COMPETENT H (DONE H ?act))) *optional
(PERSUADE H (GOAL H Ev(DONE H ?act))) *optional

Figure 1: High-level Text Plan for Recommending an Act

EFFECT: (MOTIVATION ?act ?goal)
CONSTRAINTS: ((GOAL S ?goal) and
(BMB S H (GOAL H ?goal)) and
(STEP ?act ?goal))
NUCLEUS: (INFORM S H (STEP ?act ?goal))
SATELLITE: (BMB S H (STEP ?act ?goal)) *optional

Figure 2: Text Plan for Motivation

where

`replace-1` = (replace (ACTOR user) (REPLACEE car-function) (REPLACER first-function))

This goal says that the speaker would like to achieve the state where the speaker believes that the hearer and speaker mutually believe that it is a goal of the hearer that the replacement eventually be done by the hearer. One of the plans for achieving this goal is shown in Figure 1. This plan operator has no constraints on its application. The nucleus becomes a subgoal to achieve a state in which the hearer and speaker mutually believe that it is a goal of the speaker that the hearer do this act. This subgoal will be achieved by the plan operator shown in Figure 3. RECOMMEND is a primitive speech act, so this node will be a leaf in the completed text plan.

Satellites of a plan operator capture the kind of discourse patterns that were captured in the schema-based approach and indicate other information that could be said to give a more elaborated text. In this plan operator, the first satellite corresponds to making the hearer competent to perform the act by telling him or her anything that he or she needs to know to do so. The second satellite calls for persuading the hearer to perform the act.

The satellites of the high-level plan in Figure 1 are both marked "optional," indicating that it would be sufficient to simply state the recommendation. The planner could choose not to expand any of the satellites and await feedback from the hearer. In this case, the system would simply recommend that the hearer perform the action. If the hearer is not satisfied with this text and indicates this by asking a question, such as "why?", the system can examine the recorded plan to find optional parts of the plan that were not expanded and determine if any of these could be used in answering the question. For example, the goal of persuading the hearer would be posted in response to the hearer's "why?".

Alternatively, the planner could choose to post any of the optional satellites as subgoals. In the context of this expert system, we assume that the user is competent to perform the recommended transformation. Therefore, the first satellite will not cause any text to be generated. Suppose the second optional satellite is

EFFECT: (BMB S H (GOAL S (GOAL H Ev(DONE H ?act)))
CONSTRAINTS: none
NUCLEUS: (RECOMMEND S H ?act)
SATELLITES: none

Figure 3: Text Plan for Recommending an Act

EFFECT: (PERSUADE H (GOAL H Ev(DONE H ?act)))
CONSTRAINTS: G = the set of g s.t. ((BMB S H (GOAL H g)) and (STEP ?act g))
NUCLEUS: loop for g in G
(MOTIVATION ?act g)
SATELLITES: none

Figure 4: Text Plan for Persuading by Motivating an Act

posted as a subgoal. The planner must then find rhetorical means for achieving the discourse goal "persuade." One of the plans for persuading a hearer to do an act is shown in Figure 4. This plan has a constraint which requires finding a non-empty set of goals which are mutually believed to be goals of the hearer and which the recommended act is a step towards achieving. This information may be retrieved from the expert system's problem-solving knowledge and the user model.

Constraints referring to the hearer's beliefs are checked against a user model which represents the state of the hearer's knowledge. We do not, however, wish to depend on this user model being either complete or correct. Furthermore, since we are assuming a highly interactive model of conversation in which we rely on the feedback from the hearer to indicate lack of understanding, the planner may assume that a constraint on the hearer's knowledge is true even if it is not explicitly indicated in the user model, keeping track of any such assumptions made. If the hearer indicates dissatisfaction with a text, the system can examine the recorded text plan to determine what assumptions could have been erroneous. It can then re-plan the text using a plan operator that does not require these assumptions or by providing the hearer with the additional knowledge necessary to make these constraints true. If more verbose text is desired (e.g., because the system has information that the hearer is a novice), the planner may immediately plan text to make the constraint true.

In this example, there is one goal which satisfies the constraint, namely ENHANCE-1, the goal to enhance the readability of the program. The planner thus posts a single subgoal:

(MOTIVATION replace-1 enhance-1)

Motivation is a rhetorical relation defined in RST. Note that this plan operator combined with the operator which satisfies the MOTIVATION subgoal shown in Figure 2, operationalize this rhetorical relation. We have added the necessary constraints to indicate when this relation can be used in constructing a text, indicating how and what to look for in the knowledge base in order to use this rhetorical strategy to achieve this discourse goal. For the sake of brevity, we will not go through the rest of this example in detail here. Figure 5 presents the final text plan for achieving the original high-level discourse goal. Once the text plan is completed, it is transformed into a representation suitable as input for the Penman text generation system. The text that will be generated is:[3]

> You should replace (CAR x) with (FIRST x) because that will enhance the readability of the program. To enhance the readability of the program, the system applies readability enhancing transformations. CAR-to-FIRST is a readability enhancing transformation because its left-hand-side is a function whose function name is a technical word and its right-hand-side is a function whose function name is an English word.

Note in Figure 5 that the rhetorical relations provide a context for choosing appropriate cue words to link the different parts of the text when realizing the text in English [Hov88]. For example, the two pieces of text linked with the rhetorical relation MOTIVATION, are connected with the cue word "because".

Advantages of this Approach

The Need for a Detailed Text Plan

As we have seen in the example, a text plan produced using this method provides a detailed representation of the text produced by the system, indicating how parts of the plan are related and which purposes different

[3]The implementation of the text planner is not yet complete. However, we expect to complete it by the time of the conference.

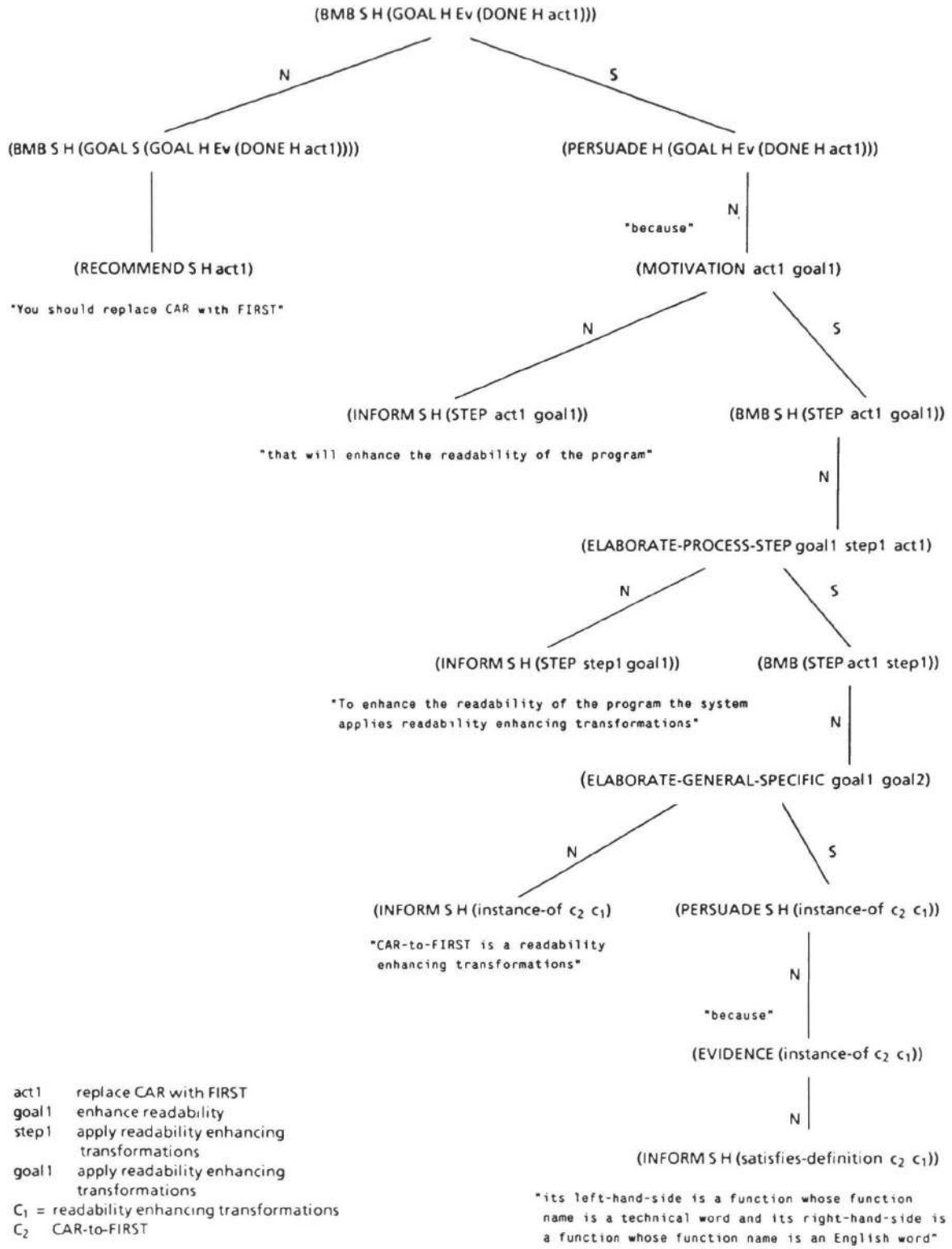


Figure 5: Completed text-plan for justification of CAR-to-FIRST transformation

parts of the generated text serve. This structure can be useful in aiding several of the cognitive tasks that have thus far been difficult for natural language generation systems, such as disambiguating follow-up questions, selecting perspective when describing or comparing objects, and providing further explanations.[4] In addition, the completed text plan can be used to update a user model upon receiving follow-up questions, as it is possible to examine the completed plan to determine whether information taken from the user model might have been inaccurate.

Unifying the Schema-based and Plan-based Approaches

Both the schema-based approach and the plan-based approach are useful in a text generation system: the plan-based approach is necessary to reason about the hearer's beliefs to produce an utterance the hearer will understand; and the schema-based approach is useful in situations where standard patterns of discourse can be identified. A generation system should thus have access to both of these kinds of text plans. In fact, instead of being two very different approaches, these two approaches should be two extremes of a continuum, where a schema can be viewed as a compiled version of a text-plan, with all the non-terminal nodes pruned out and only the leaves remaining. They can produce the same behavior, but all of the rationale for the behavior has been compiled out.[5] Because of this "compilation," a schema is computationally efficient. It may be necessary however to recover the information which has been compiled out in order to handle some discourse situations.

We believe that the approach proposed in this paper provides a computational model that unifies the schema-based and the plan-based approaches. The plan structure proposed can express the whole spectrum of plans: from primitive plans for speech acts to plans that achieve very high-level discourse goals and resemble schemata. Once a plan for achieving a high-level discourse goal has been constructed, it will be possible to use its "compiled" form (just the leaves) for the sake of computational efficiency. By keeping the detailed plan that resulted in this schema, the system will still be able to recover from failure. Most plans, however, like the ones presented in our examples, can be considered as a middle ground between the schemata and the plans that were used in previous plan-based approaches.

Future Work

Many issues remain to be addressed. In particular, we must develop a set of heuristics for plan selection as well as for deciding when an optional satellite should be expanded. We must also devise a control mechanism for these heurictics.

We have mentioned that the rhetorical relations used to form a plan can provide information as to which cue word is appropriate. Rhetorical relations may relate paragraphs as well as simple propositions. The planner must recognize this fact and act upon it in the appropriate way to ensure that the resulting text is easy to read: for example, the current focus might have to be reintroduced before a new proposition is presented. We will investigate the use of *critics* that will examine a plan for global interactions and make decisions about how to realize a large text plan in understandable English.

We are also investigating the possibility of remembering "good" plans, namely plans that were successful in presenting the information to the user without triggering a flock of follow-up questions. In a similar vein, we are looking into "learning" plans for achieving high-level discourse goals, based on the information presented to the user and follow-up questions asked. We would like the system to be able to learn schemata that are used often.

Conclusions

In this paper, we proposed a computational model for text planning which relates effects on the hearer's beliefs to the rhetorical means that can be used to achieve these effects. As a result, we are operationalizing Rhetorical Structure Theory for constructing text. We have argued that a detailed text plan which explicitly records the rhetorical means used to construct the text is necessary in order to respond to follow-up questions. The text planner proposed can be used to generate either verbose texts or answers to follow-up questions, in a more interactive situation.

[4] See [MS88] for more details about a system that uses such a plan to do these tasks.

[5] Note that this is similar to Swartout's argument in the XPLAIN system [Swa83], where Swartout points out that having implicit knowledge is usually not enough: one might need to recover the knowledge omitted in a compiled form.

References

[AP80] James F. Allen and C. Raymond Perrault. Analyzing intention in utterances. *Artificial Intelligence*, 15:143–178, 1980.

[App81] Douglas E. Appelt. *Planning Natural Language Utterances to Satisfy Multiple Goals.* PhD thesis, Stanford University, 1981.

[Aus75] J. L. Austin. *How to Do Things With Words.* Harvard University Press, Cambridge, MA., 1975. Second Edition.

[CL85] Philip R. Cohen and Hector J. Levesque. Speech acts and rationality. In *Proceedings of the Twenty-Third Annual Meeting of the Association for Computational Linguistics*, pages 49–60, University of Chicago, Chicago, Illinois, July 8-12 1985.

[CP79] Philip R. Cohen and C. Raymond Perrault. Elements of a plan-based theory of speech acts. *Cognitive Science*, 3:177–212, 1979.

[Gri75] Joseph E. Grimes. *The Thread of Discourse.* Mouton, The Hague, Paris, 1975.

[Hov88] Eduard H. Hovy. 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, 7-10 June 1988.

[McC85] Kathleen F. McCoy. *Correcting Object-Related Misconceptions.* PhD thesis, University of Pennsylvania, December 1985. Published by University of Pennsylvania as Technical Report MS-CIS-85-57.

[McK82] Kathleen R. McKeown. *Generating Natural Language Text in Response to Questions About Database Structure.* PhD thesis, University of Pennsylvania, 1982. Published by University of Pennsylvania as Technical Report MS-CIS-82-5.

[MM83] William C. Mann and Christian Matthiessen. *Nigel: A Systemic Grammar for Text Generation.* Technical Report RR-83-105, USC/Information Sciences Institute, February 1983.

[MS88] Johanna D. Moore and William R. Swartout. A reactive approach to explanation. 18-21 July 1988. To be presented at the Fourth International Workshop on Natural Language Generation.

[MT87] William C. Mann and Sandra A. Thompson. Rhetorical structure theory: a theory of text organization. In Livia Polanyi, editor, *The Structure of Discourse*, Ablex Publishing Corporation, Norwood, N.J., 1987. Also available as USC/Information Sciences Institute Technical Report Number RS-87-190.

[NSM85] Robert Neches, William R. Swartout, and Johanna D. Moore. Enhanced maintenance and explanation of expert systems through explicit models of their development. *IEEE Transactions on Software Engineering*, SE-11(11), November 1985.

[Par87] Cecile L. Paris. *The Use of Explicit User Models in Text Generation: Tailoring to a User's Level of Expertise.* PhD thesis, Columbia University, October 1987.

[Sac75] Earl D. Sacerdoti. *A structure for plans and behavior.* Technical Report TN-109, SRI, 1975.

[Swa83] William R. Swartout. XPLAIN: a system for creating and explaining expert consulting systems. *Artificial Intelligence*, 21(3):285–325, September 1983. Also available as ISI/RS-83-4.

DEFEASIBILITY IN CONCEPT COMBINATION: A CRITERIAL APPROACH

BRADLEY FRANKS TERRY MYERS SCOTT McGLASHAN
CENTRE FOR COGNITIVE SCIENCE
UNIVERSITY OF EDINBURGH

CONCEPT COMBINATION AND DEFEASIBILITY

What are the semantic relations within and between lexical concepts that constrain their combination? The position advanced is that they are criterial relations. Such a view has advantages over extensional and standard intensional approaches, which fail to capture adequately the characteristics of defeasibility at play during such combinations. We will examine these questions with respect to the adjectival modification of nouns.

Approaches To Concept Combination

Treatments of concept combination divide into "extensional" and "intensional" theories. The major failings of both approaches can be traced to some shared assumptions. Extensional approaches include set-theoretic treatments of concept combination: concepts are treated as unanalyzed units, and the detailed interactions of the contents of the concepts are bypassed. "Intensional" views construe concepts according to their internal properties, and concept combination is reduced to the interactive mechanism for the inheritance of properties from the inputs. Most writers postulate some partitioning of the features into, for example, "necessary" and "characteristic" types: in combinations, the "necessary"/deductive attributes of both concepts are to be inherited by the conjunction, whilst the "characteristic" attributes might be negotiated.

The shared assumptions of the two views are as follows. Firstly, the crucial attributes are those which are deductively related to the concept: as Jackendoff (1983) has pointed out, a defining-features intensional account is formally equivalent to a meaning-postulate extensional account. Secondly, and linked to this: the semantic object is a "total object" (Landman, 1986). Total objects are crucial to classical logic and realist semantics. An object is equated with the set containing *all* of its properties: these properties are the deductively-related identity conditions. Total objects are thus predicated upon the assumption of total evidence or information, even though this may be beyond the grasp of a particular language-user. Thirdly, neither view can distinguish between the conditions of application of

a concept, and the content of that concept. Since they are (total) object-centred, the assumption is that the concept can be applied only if all of its identity conditions are fulfilled; and these are usually taken as conditions on being a particular kind of object.

Standard intensional approaches have a further quality which is founded upon the above three assumptions. The characteristic properties are construed as default properties. And an object-centred approach demands that where the characteristic features of a concept obtain, so must the necessary features.

The major difference between the extensional and intensional approaches is that the former takes the necessary attributes and operates over them as a set, in terms of formal operations only. The intensional approach operates upon the attributes themselves, allowing the further use of non-necessary attributes.

The Criterial Relation

The criterial relation (henceforth, "C-relation") is a semantic relation which may hold either between two different concepts, or between a single concept and its evidential conditions. The notion derives fairly directly from the later Wittgenstein (1953). Traditionally, criteria are said to have the following properties. Firstly, the relationship between the evidence (qua criteria) and the particular claim (qua application of a concept) is somewhere between deduction (since it confers certainty), and induction (it is defeasible). Secondly, as the criteria are necessarily good evidence for a claim, they fix the semantic content of that claim. Thirdly, to have satisfaction of the criteria for a claim is consistent with obtaining further information which overturns that claim. The relation is thus defeasible. Fourthly, criteria are generally held to be multiple. In addition to criteria a concept has symptoms, or S-relations. These are inductively related features, similar to default properties in that their defeat will not alter the identity of the particular claim. So both semantic relations are inherently defeasible.

This reflects the received criterial view. The approach adopted here extends the concepts according to an anti-realist semantics. Primarily, C-relations govern the conditions of application or use of a term, rather than conditions on identity of an object (cf. Tennant, 1987). The criteria for a concept are C-related to the mental representation of the object, which is itself S-related and C-related to its attributes, and to other concepts. C-relations are the conventionally underpinned semantic relations holding between the evidence for, and the application and evaluation of, a concept. Hence, C-relations embody partial objects based upon partial information: they concern how we may reliably "go beyond the given" evidence to infer a particular kind of conceptual entity. C-relations license defeasible

extensions of partial objects to less partial ones.

This allows us to specify the major difference between S-relations and default properties; since the properties are not tied to the identity conditions of objects, where the S-relations of a concept obtain, there is no necessity that the C-relations do so. Hence, a concept may be applied on the basis of S-relations alone, but with less warrant than a C-related application.

The final difference between the standard views and the one adopted here is that the central semantic relation is content-driven and non-transitive. This issues in a type of defeasibility which is not open to the other views.

Types of Defeasibility

Three kinds of defeasibility are pertinent: rebuttal, undercutting and default. The first two involve inter-conceptual conflict and defeat, the last one intra-conceptual. But firstly, we define a

Defeater: if P is evidence for Q, R is a defeater for this evidence iff:

R is consistent with P; and

(P & R) is not evidence for Q.

Crucially, anti-realism's denial of excluded middle denies that this is equivalent to evidence for (-Q).

Rebuttal: Denial of the Claim:

Rebutting Defeater: If P is prima facie good evidence for Q, R is a rebutting defeater for this iff:

R is a defeater; and

R is good evidence for (-Q).

Now, there are two types of rebutting defeaters:

Type I: R is a type I rebutting defeater iff:

R is a rebutting defeater; and

R is good evidence to support the claim that P would not be warrantedly assertible unless Q were so. Hence, (-P).

(The second clause is, of course, a generalization of the definition of modus tollens in classical logic: -P unless Q).

Type II: R is a type II rebutting defeater iff:

R is a rebutting defeater; and

R is good evidence to deny the claim that P would not be warrantedly assertible unless Q were so. Hence, P may be assertible.

(Here, the second clause generalizes the denial of modus tollens).

Any defeasible deduction will necessarily be a type I rebutting defeater, since modus ponens supports modus tollens. Hence, set-theoretic and standard intensional views can utilize only this type. How would this operate? A defeasible concept is applicable iff we can show that *none* of the defeating conditions obtains. So certainty is equated with necessity, and necessity is read as a deductive relation. Any weaker relation is inherently doubtful. The logical possibility of doubt in any situation is equipollent with an actual grounded doubt in the current situation.

Undercutting: Denial of the C-relation:

Defeat of C-relations occurs by an:

Undercutting Defeater: If P is prima facie good evidence for Q, R is an undercutting defeater for this iff:

> R is a defeater (not a rebutting defeater); and
>
> R is a good evidence to deny the claim that P would not be warrantedly assertible unless Q were so.

(Note, in this case, the second clause cannot be a generalization of modus tollens, since we do not have an outright denial of Q).

This type of defeater, then, attacks the *connection* between P and Q rather than Q itself. The defeater is a reason for denying that we would not have the evidence unless the conclusion were true. This does not imply either -Q or -P: it denies that a claim or generalization (Q) on the basis of P would be warrantedly assertible. But the evidence itself is still assertible, and indeed the claim could still be so.

The epistemological assumptions upon which C-relations are predicated are a great deal more naturalized than their classical realist counterparts (Baker & Hacker, 1984). The obtaining of a C-relation provides certainty in the sense that it is conclusive evidence for the claim. And it is conclusive in the sense that the evidence cannot be improved upon, even if it is multiplied: and it can nonetheless be overturned. The burden of proof is also altered: on the classical view, there is a *requirement* to check and deny the (possibly open-ended) list of potential defeaters. On an anti-realist view, there is no such requirement: if the prima facie evidence supports the claim, then, if there are no available prima facie defeaters, the claim will go through - even though there is a possibility of defeat. So the logical possibility of doubt is not equivalent to the existence of an actual grounded doubt.

Default: Denial of Inductive Properties:

This is the approach exemplified in the S-relation and utilized in frame and prototype theory. A default property is either a parameter or a value of a parameter which is linked to a concept by an inductive

relation: it is a "typical" property. Where the property is overridden, the application of the concept itself is nonetheless still justified.

ADJECTIVE-NOUN COMBINATION

Noun Phrase Constructions

One type of construction which illuminates the different kinds of defeasible semantic relations are NPs where an A modifies the N, as in sentence-type 1, below; the crucial questions concern how this modification operates, especially in cases of conflict of A and N properties, and whether we can derive sentence-types 2 and 3 from the type-1. The sentence-types can be rendered:

1: This is a (A)(N) 2: This is A 3: This is a N

It will be evident that the derivation of type-3 sentences is equivalent to the derivation of the superordinate category of the N from that of the NP, with the preservation of the sortal type. For *privative* adjective-types, the inferences to type-2 and type-3 sentences are problematic: the A functions neither predicatively nor attributively. Consider the following case:

1: This is a fake Hogbin 2: This is fake 3: This is a Hogbin

Here, we cannot infer 3 from 1, although 2 is sensical. In terms of classical logic's set-theoretic treatment, privatives are modelled by a condition which requires that, if X is a member of the set of entities denoted by N, it is a member of the relative complement of the NP set, in the particular domain. (Hoepelman, 1983).

Now, the kinds of examples with which we are concerned are ones which we term "functional privatives": ones where the inference to type-3 sentences is made problematic by the *interaction* of the semantic properties of the N and A. Thus functioning privatively is *not* an intrinsic property of the type of adjective being used - rather, it stems from the conflict of semantic features and the principles of defeasibility by which they are resolved. Examples include:

This is a stone lion

This is a plastic flower

These adjectives can function affirmatively in other contexts:

This is a stone bridge

This is a plastic chair

They each allow inferences to their appropriate sentence types 2 and 3. In the functionally privative cases, the inference to type 3 is problematic, although not completely nonsensical.

Problems With The Deductive Approach

If we map a deductive relation over total objects, the only form of defeasibilty open is type I rebuttal. So the denial of the evidence for the necessary conditions of a concept entails a denial of all of its conditions - i.e., of the concept itself. This is precisely the effect achieved by the set-theoretic relative complement method. Three problems emerge: firstly, it denies too much: with functional privatives, we would like to preserve at least some of the non-essential properties of the "lion" concept; since they support modus tollens, and require default properties to be dependent upon the obtaining of necessary conditions, standard views cannot allow this. Secondly, there is no account of the asymmetrical nature of the combination. We can argue that this will largely be at the behest of dependency relations, in which the C-relations of the A have primacy (cf. Anderson, 1986). Thirdly, the method can only tell us what the object *is not* rather than what it is: a "fake Hogbin" just is not a Hogbin, and a "stone lion" is not a lion: yet it seems that a "fake Hogbin" is somewhat more like a real Hogbin in appearance than is a Rothko. Again, a stone lion is more similar in shape to a lion than to a frog. The argument of the next section will be that the C-relation view can make good most of these flaws.

THE CRITERIAL APPROACH

We claim there are two different defeats at work here: one type II rebuttal, the other an undercutting. Ordinarily, a C-information based concept of "lion" will function as evidence for the sortal identity of an object: the conceptual evidence (P) is C-related to the identity claim (Q). But the combination of the concept "stone" (R) with P will undercut that C-relation. The combination of P with R involves type II rebuttal of the C-information lodged in P by the C-information in R. So, concept Q's use is C-related to certain kinds of information which is generated in support of the use: for example, ideas of "internal essence" of the identity of lions, derived from conventionally structured lay theories of the domain (see Murphy & Medin, 1985; Keil, 1987). If these properties were necessary/deductive conditions on a total semantic object, the overriding of them would function according to rebuttal type I; hence the term could not be applied. That this is not the case is clear from the ease with which we can understand "stone lion". The concept's use is also S-related to certain features, concerning "appearances". So we can justifiably utilize the term "lion" if we can generate appropriate C- and S-relations (i.e., P) on the basis of current context and information. But we have an undercutting of the connection between P and Q, by R: that is, the applicability of the term "lion" on the basis of P is denied. This denial of positive support for Q is not equivalent to an assertion of Q's falsity. And this also leaves P open to negotiation.

For undercutting, the combination (P & R) must be consistent; this is where the type II rebuttal of the noun's C-information, which motivates the undercutting, operates. Crucially, this type of defeat could

only be facilitated by a context of an undercutting, and not by a rebuttal of claim Q. The question is then, which has priority, P (the noun), or R (the adjective)? On the basis of linguistic dependency, the C-relations of "stone" (such as "inanimate", etc.) have the effect of overriding the C-relations of "lion" by a type II rebuttal. The full concept of "stone" is thereby rendered consistent with the undefeated S-relations of "lion". Two points are noteworthy here. First, the survival of some of the S-information is acceptable since S-relations do not require the obtaining of C-relations. Second, in this combination not all of the S-information survives: only that which is *required* by the C-relations of the "stone" concept. The other S-relations would either be directly negated (by default defeat), or rendered inconsistent (by general defeat). But the C-relations of "stone" positively require that there is some physical structure or shape: and it is these qualities of "lion" which are retained. We claim that the defeat of the C-information of "lion" by that of "stone" is type II rebuttal because it is a cancellation of the criterial *properties* of the noun; and the other defeat is by undercutting because it defeats a C-related *claim* made on the basis of such properties.

We have advocated a well-motivated intensional account of the defeasible semantic relations constraining concept combination. Such an account seems to be necessitated by the case of functional privatives. Its extrapolation to more straightforward concept combinations should be perspicuous.

REFERENCES

Anderson, J. (1986), Suprasegemental dependencies, in J. Durand (ed.), *Dependency and Non-Linear Phonology*. London: Croom Helm.

Baker, G.P., & Hacker, P.M.S. (1984), *Scepticism, Rules and Language*. Oxford: Blackwell.

Hoepelman, J. (1983), Adjectives and nouns: a new calculus, in R. Bauerle, C. Schwarze & A. von Stechow (eds.), *Meaning, Use and Interpretation of Language*, Berlin: de Gruyter.

Jackendoff, R. (1983), *Semantics and Cognition*. Cambridge, Mass.: MIT Press.

Keil, F.C. (1987), Conceptual development and category structure, in U.Neisser (ed.), *Concepts and Conceptual Development*. Cambridge: CUP.

Landman, F. (1986), *Towards a Theory of Information*. Groningen-Amsterdam Studies in Semantics, 6, Dordrecht: Foris Publications.

Murphy, G.L., & Medin, D.L. (1985), The Role of theories in conceptual coherence, Psychological Review, 92 (3), 289-316.

Tennant, N. (1987), *Anti-Realism and Logic*. Oxford: Clarendon Press.

Wittgenstein, L. (1953), *Philosophical Investigations*. Oxford: Blackwell.

THE COMPREHENSION OF ARCHITECTURAL PLANS BY EXPERT AND SUB-EXPERT ARCHITECTS

Janice D. Gobert & Carl H. Frederiksen

McGill University

While much research on text comprehension has focused on narrative text, recent research has examined the underlying conceptual network representations and processes involved in the understanding of particular types of text structures such as narratives, procedures, conversations, problems, and descriptions (Frederiksen, 1985). One particular type, namely descriptive text, is used to present information about objects, states, events, or processes to a learner. The information given in a descriptive text allows the reader to develop networks of descriptive semantic information about a given topic. The focus of interest in studies such as these is to evaluate both the amount and type of descriptive semantic information which the learner has acquired from the text and the rules that are used to generate and integrate conceptual structures.

However, one may also acquire descriptive semantic information about an object from information sources other than text. That is, descriptive semantic structures are not language bound. There exist information sources which are non-linguistic and from which one also can generate a semantic description of an object. The ability to assign meaning to an object, regardless of the medium of presentation, is an important comprehension ability which takes place in everyday life.

Important domains in which information is presented in non-linguistic form are those that deal with graphics. Like texts, graphics present information about an object, but they do so in a different way. Thorndyke and Stasz (1980) have defined one type of graphic, namely, a map as a "symbolic 2-dimensional representation of an area which is large enough to navigate, i.e., a building, a city, or a country." Furthermore, maps are differentiated from other learning materials in two ways. First, they are more complex than typical textual materials because they contain both spatial and conceptual information, and they represent characteristics such as shape, and absolute and relative positions. Secondly, they present all their information simultaneously. Thus, in tasks in which graphics are used as sources of information, the ability to comprehend or interpret these sources is essential.

The purpose of the present research was to examine the underlying processes and representations involved in the comprehension of graphic information sources in architecture. The working hypothesis is that semantic processes and representations which are used in the understanding of descriptive texts should also operate in the understanding of graphic information sources. Architecture is an interesting domain in which to study this for two reasons. First, graphics are the information sources which are most commonly dealt with, and secondly, architects are trained to give verbal descriptions of buildings from their various drawings. This is a crucial and highly developed skill in architecture, and plays an important role in the evaluation of architecture students. During this evaluation session, called a Critique, students must give a full description of the building which they have designed, to a committee of examiners. Since this skill is part of the professional training of architects, the task of generating a verbal description based on the drawings of a building is not one which interferes with the processing or comprehension of the graphic stimuli themselves (cf. Ericsson and Simon, 1980). Furthermore, since the output is verbal, it allows for the application of methods of semantic analysis developed for natural language to the verbal protocols produced by subjects. These techniques of propositional and conceptual frame analysis

(Frederiksen, 1975, 1985) may be used to abstract the semantic information from subjects' natural language protocols and are both well-defined (in terms of semantic BNF grammars) and rigorous (analysis is guided by a computer program). Thus, they provide an excellent means to ascertain both the amount and type of information an individual has learned about a building from its graphic sources.

It is important to note that the problem that we are interested in is not one of how graphic information is perceived or encoded; rather, how its meaning (i.e., a description of the object it represents) is represented in "think aloud" protocols given by the subjects. Therefore, the nature of memory representation for visual objects is not an important issue here since the task was a "think aloud" interpretation of the object represented by the graphics, rather than of the graphic information itself. Of course representations for visual objects having characteristics which are symbolic and spatial in nature, are generated. Furthermore, these are required when modelling the processes by which semantic interpretations are assigned to graphic objects. However, since our current interest is in the semantic representations of the objects being represented by graphic information, the debate over the nature of representation for visual objects (cf. Anderson, 1977) will not be discussed.

The present research has several objectives. Perhaps the most important of these is to examine the processes involved in comprehending a chosen building from its graphic sources; that is, how do architects construct a "mental model" (Johnson-Laird, 1980) of a building as a three-dimensional entity from its graphics which present the information schematically in two-dimensions. Comprehension of graphics in architecture involves the interpretation of the graphics with regard to the following types of conceptual models: (a) the descriptive properties of the building, (b) its supporting structure, (c) its geometry, (d) possible movement and circulation in the building, (e) functions of building components and spaces, as well as other characteristics including its goals and its interpretation by a potential user of the building. In addition, any of these conceptual models can involve different categories of semantic information which are used to represent the meaning of graphic displays by architects. Semantic categories include the following which are identified in propositional models: categorization, attribution, function, composition (part structure), locative information, identity, similarity relations, events involving building components (e.g., movement), and algebraic, and dependency relations.

A second objective is to identify from the verbal protocols characteristics which differentiate expert and sub-expert performance for this task based on their semantic interpretations of the chosen building.

A third objective is to investigate the nature of expertise in architecture which is a domain that is both highly symbolic and semantically rich.

The most important previous work within the domain of architecture pertinent to the present study is that of Akin (1979) who has adopted a problem-solving approach to design in architecture. Akin has studied the heuristics used in the design process. However, the semantic representation and interpretation of the graphic information is not addressed centrally in his work. Thus, the present research will contribute to the body of expert-novice literature with architecture as the domain of study. It also has the potential to contribute information and methods pertinent to other domains in which the sources used are primarily graphic or pictorial in nature, such as cartography and radiology.

Method

Subjects

A total of eighteen subjects participated in this study. Seven of these, the "sub-expert" group, were students who had recently completed their fourth year in the school of Architecture at McGill University. A total of nine professional architects comprised the expert group, all of whom had

a minimum of two years experience as professional architects.

Two professors in the School of Architecture at McGill University also participated in the study for the purposes of developing an expert descriptive frame, i.e., a Reference Model for the building. These experts were chosen for their particular expertise pertinent to the task of providing a verbal description of buildings based on graphic information. One has particular expertise in the analysis of building plans and the verbal description of such plans; the other has developed a classification system for the characterization of architectural information systems which is used to teach students how to interpret buildings. These two experts will be referred to as "participant experts".

Materials

Four plans (ground , second level, third level, and roof) , three sections, an axonometric drawing and two aerial photographs of this building comprised the materials for this study. The building which was chosen for this study is The Atheneum, which was designed by Richard Meier. This building, which is located in New Harmony, Indiana, serves as an information and tourist centre for the towns' visitors. It was selected for its interest and the availability of these particular graphic information sources for it.

Procedure

The graphic sources were presented in the following order: plans, sections, axonometric drawing, and aerial photographs. Each set (plans, sections, axonometric, and aerials) presented additional information about the building. The procedure for testing was identical for all participants. All subjects were asked to give a "think aloud" interpretation of the sources. More specifically, they were asked "to describe, in as much detail as possible, all the information that they knew about the building from the sources given, specifying which source they were referring to, and the location within it (top, bottom, left, right, etc.)". All protocols were tape recorded and transcribed. All plans were available simultaneously for examination by the subjects, and subsequently all sections were simultaneously available. Thus a subject could shift from one graphic source to another of the same type during the performance of the task.

The present study will analyze only the results from the first type of graphic source given, namely, plans. Since the other sources were introduced after these, the subsequent information sources cannot have had an influence on the interpretation of the building plans.

Development of Reference Model from Participant Experts' Protocols

The coding of subjects' protocols was based on matching them to a Reference Model of the building. To develop such a model, protocols produced by the participant expert architects were analyzed. This involved several steps. First, all statements in their protocols which were descriptions of the building were identified (in contrast to statements such as architectural critiques, historical influences, descriptions of one's thought processes, etc.). The purpose to this was to have protocols for propositional analysis which would include only descriptive information about the chosen building.

The second step was to identify all graphic objects which were referred to by the two participant experts and assign them alphanumeric codes. Thus, each graphic object was assigned a numerical label for identification purposes.

Third, a propositional analysis (Frederiksen, 1986) was carried out on each of the expert protocols in order to ascertain the semantic information described by each expert.

Fourth, a descriptive semantic network (i.e., descriptive frame) was constructed based on these propositions. Finally all links (i.e., correspondences) between semantic objects in the network and graphic objects were identified.

Coding of Subjects' Protocols

Subjects' protocols were transcribed,

descriptive statements were identified (as in the participant experts' protocols), and these statements were matched to information in the Reference Model and to graphic objects when reference was made to graphic elements. The basic data resulting from this coding of subjects' protocols is the presence of specific nodes or links of the Reference Model, of graphic objects for which semantic descriptions were made, and of links between semantic information and graphic objects. From these basic data, frequency counts were made of the various categories of semantic or graphic information.

Categories for Data Analysis

Graphic categories. These include the number of graphic objects from each plan that are linked to semantic descriptions, and the frequencies of particular types of graphic information such as: objects, structures, spaces, features, etc.

Types of descriptive models. These include the amount of semantic information included in subjects' interpretation protocols that reflects different types of models of the building, in particular: building description, supporting structure, geometry, and circulation and access.

Categories of semantic information in the network. Categories of semantic information with examples taken from the participant expert's protocol were as follows:

1) object classification or identification: this category consists of objects whose identity or category was given. Examples of this are:

"That's probably the space [over the auditorium]", "This must be the balcony [here, over the stage]".

2) attribute description: descriptive information given with regard to the physical attributes of the objects. Examples are:

"[It's] a pretty high semi-enclosed space", "It's a pretty odd building."

3) function: includes information with regard to the function of objects within the building, and the functions of the building itself. Examples are:

"[It doesn't seem like it's a theatre of sorts], more for music I imagine", "[It's a ramp] for the handicapped".

4) information about events, i.e., circulation and movement through the building: refers to statements given with regard to the paths travelled by people who are in the building. Examples are:

"This is the top level which gives access to this stair", "you can go from this level higher up to the 3rd floor".

5) physical processes underlying building structure: information pertaining to the physical structure and processes of the building or it's various parts. Examples are:

". . .which is supported by this column here". "Inside this space which is used to support this 3rd floor part of the building which sticks out."

6) composition (part structure): information referring to the various parts of the building. Examples are:

"it is a building that has a large auditorium [which is for public access]", ". . it seems to have larger and smaller elements. . ."

7) Point location: i.e., an object which goes from one point in a 1, 2, or 3 dimensional region to another point in a 1, 2, or 3 dimensional regional, or an object which goes from one point in a 1,2, or 3 dimensional region but has no specified secondary location. Examples are:

"This is a stair which you can go from the front.", " It goes from the ground floor to the 2nd floor, and ends up at the 3rd floor."

8) Containment location: the location of an object which is contained in a 1,2,or 3 dimensional region. Examples are:

"which has a stair which should show up at the second floor", "It's outside the building envelope essentially".

9) Adjacency location: location which is specified as being relative to the location of another object, i.e. below , above, etc. Examples are:

"That's probably the space over the auditorium", "There's a space underneath here."

10) Direction location: location which is also specified as being relative to a centre point, or location which specifies the direction which an object goes toward. Examples are:

"The space which is on the north side.", "These stairs are going up".

For further definition of types of location see Frederiksen (1975).

Links between graphic objects and semantic descriptions. Categories are based on combinations of types of graphic objects and types of semantic information associated with them in subjects' protocols. For example, graphic objects may be assigned to semantic categories, spaces may be assigned functions, and particular features may be associated with relative locations.

Design and Construction of Within Subject Variables

The design of the study involves one between-group factor: level of expertise. Within-group factors are constructed to correspond to the above mentioned categories. In each of a series of multivariate repeated measures analysis of variance, particular within-group factors will be investigated in terms of their main effects and their interactions with the between-group factor (i.e., to investigate expert and sub-expert differences in terms of the effect).

Results and Discussion

Results in this paper will be restricted to illustrative data for one of the participant expert architects and two of the sub-experts.

Frequencies were tabulated for all of the categories of semantic information (objects, descriptions, function, circulation, structure, composition, and the four location categories, namely, point location, containment location, adjacent location, and direction location) for the participant expert and each of the two sub-experts. These categories of semantic information were grouped into three sets since they reflect different aspects of the subjects' interpretations of the plans.

The first set of measures consisted of frequencies for two categories: the number of objects identified in the building, and the amount of descriptive information given about the objects in the building. Results are given in Figure 1. Here we see that the participant expert produced higher frequencies than did the sub-experts for both these categories. Thus, the participant expert identified a greater number of objects within the building than did the sub-experts, and he gave more descriptive information about the building as well.

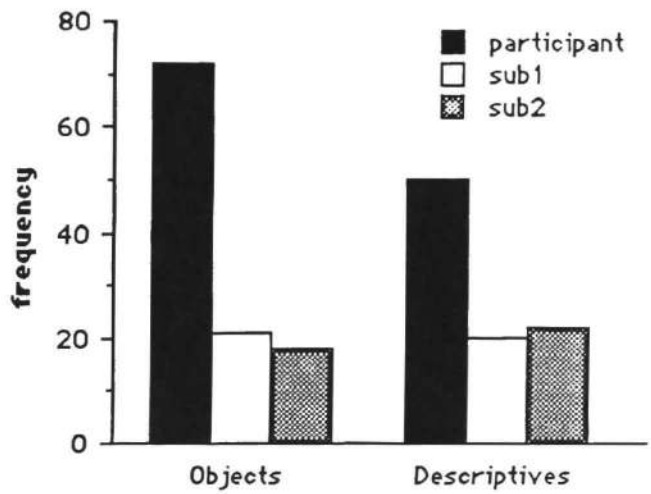


Figure 1: objects & descriptives

The two sub-experts produced frequencies which were very similar to each other for both of these categories; however, both the sub-experts gave slightly more descriptive information than object identity information. This is in contrast to the participant expert, who gave a larger amount of object identity information than descriptive information. Thus the expert's model of the building included more objects and a greater amount of descriptive information for these objects. The sub-experts' identified fewer objects, but identified a greater amount of descriptive information for these objects.

The second set of frequencies consisted of the following categories: function, geometry, circulation, structure, and composition/part. Figure 2 presents these results. The participant expert

produced higher frequencies than did the sub-experts for three of the five categories, namely, function, circulation, and part/ composition. Sub-expert 1 produced the highest frequency for geometry, whereas the participant expert produced the lowest frequency for this category. With respect to structure, the participant expert and sub-expert 2 produced the same frequencies.The two sub-experts produced very different patterns of frequencies.

Since these measures reflect the extent of particular types of conceptual information in the "Building Models" constructed by the subjects, differences in frequencies of a category may reflect either differences in ability to generate particular kinds of semantic information, differences in strategies reflecting the strategic importance assigned by a subject to particular kinds of semantic information, or both. For example, if we assume that differences in frequencies across categories for the Expert reflect strategic differences (and not ability differences), we may conclude that the expert gave strategic priority to: (a) functions of building spaces and components, (b) circulation or movement through the building (what we may call the "Walk Through Strategy"), (c) the composition of the building, that is its part structure, and (d) the supporting structure of the building. The expert did not give priority to the geometry of the building as it was reflected in the plans.

Sub-expert 1 gave priority to the geometry of the plan, adopting a "geometric strategy", and to the part structure, i.e., the composition of the building. This subject also appears to have used the "Walk-Through Strategy". The second sub-expert also emphasized geometry, but also generated information in all of the other categories, with the exception of circulation. Thus, this subject appears to have adopted strategic priorities more like those of the expert, but was less successful in applying them. It is interesting to note that the "Geometric Strategy", unlike the other strategies, involved principally a description of the plans rather than the building which is represented by the plans. Sub-experts apparently carried out "shallower" processing of the plans, emphasizing information which was more directly depicted in the plans.

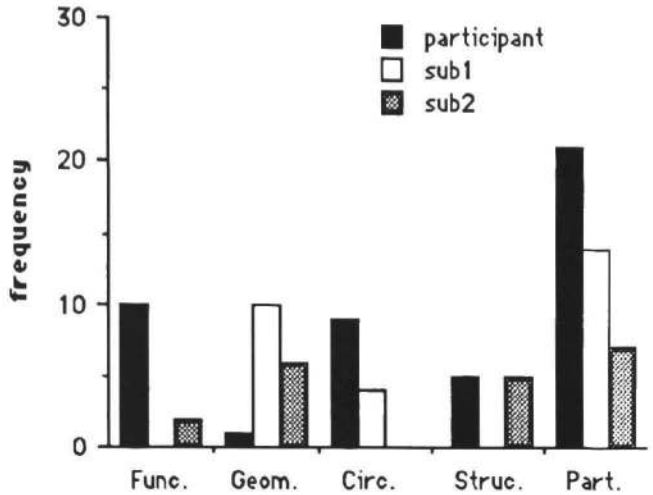


Figure 2: categories 3-8

The third set of frequencies comprises the four locative categories identified previously. Differences in the frequencies of these categories reflect the types and complexity of locative descriptions of the building. Figure 3 presents these results. The participant expert produced high frequencies for three of the four locative categories: point location, containment and direction. Containment location involves specifying inexact location of an object with reference to a region of the building (e.g., a floor or area). Point location involves more precise location of objects (or parts of objects such as their extremities) with respect to other objects. For containment location, the frequencies of the two sub-experts were identical and much lower than those of the expert, while for point location, the sub-experts' frequencies were lower than that of the expert and were different from each other: sub-expert 2 was very low and sub-expert 1 was intermediate in frequency.

With respect to the other two locative categories, adjacent location and direction, differences between the expert and sub-expert 1 were less apparent. Sub-expert 1 showed higher frequencies than did either the expert or the other sub-expert for adjacency location For direction location, the frequency for the expert was very similar to that produced by sub-expert 1. Sub-expert 2 produced

frequencies that were very low for both of these categories. Except for containment location, the expert and sub-expert 1 were similar in the types of locative structures they generated to represent the spatial characteristics of the building, while sub-expert 2 had difficulty generating a spatial-locative representation for the building.

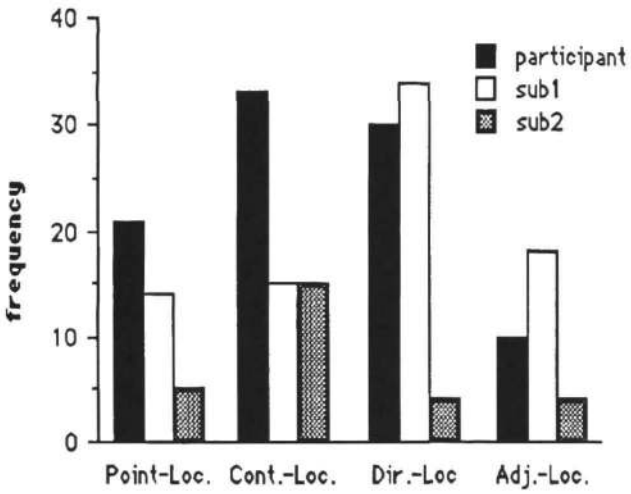


Figure 3: location

Although these results are only preliminary, they illustrate the kinds of differences that an analysis of the complete data on these measures is likely to reveal. The effects of prior knowledge of the building will also be analyzed in order to see whether there are any significant effects on these categories. It is interesting to note that many of the sub-experts in the full sample had some prior knowledge of the building due to the fact that it is a modern one and modern architecture comprises part of their curriculum; however, preliminary analyses suggests that the performance for these subjects on several of the categories was still lower than that of one participant expert architect who had no prior knowledge of the building whatsoever.

Preliminary analysis of these data suggest that the processes involved in comprehending architectural drawings are different, both quantitatively and qualitatively, for experts than they are for sub-experts. These differences, if substantiated by analyses of variance, may be attributed to differences in strategies employed in generating a semantic description of the building, and in the effectiveness of the processes associated with different categories or types of descriptive. In addition, prior knowledge of engineering, building functions, principles for structuring space, geometric principles used in design, and principles governing movement in architectural spaces are certainly involved. What is clear, however, is that very complex and differentiated semantic information structures are involved in the comprehension of architectural plans, and that these structures can be uncovered through a semantic analysis of natural language interpretations.

REFERENCES

Akin, O. (1979). Models of architectural knowledge: An information-processing view of design. Dissertation Abstracts International, 41, 3, p. 833-A. (University Microfilms No. 72-8621.)

Anderson, J. (1977). Arguments concerning representations for mental imagery. Psychological Review, 85, 249-277.

Ericsson, K. & Simon, H. (1980). Verbal reports as data. Psychological Review, 87, 215-251.

Frederiksen, C. (1975). Representing logical and semantic strucutre of knowledge acquired from discourse. Cognitive Psychology, 7, 471-458.

Frederiksen, C. (1985, June). Comprehension of different types of text strucutre. Paper presented at the Annual Meeting of the Canadian Psychological Association, Halifax, Nova Scotia, Canada.

Frederiksen, C. (1986). Cognitive models and discourse analysis. In C. Cooper, & S. Greenbaum (Eds.), Written communication annual volume 1: Linguistic approaches to the study of written discourse, Beverly. Hills, CA: Sage.

Johnson-Laird, P. (1985). Mental models. In A. Aitken, & J. Slack (Eds.), Issues in Cognitive Modelling, London, England: LEA publishers.

Thorndyke, P. & Stasz, C. (1980). Individual differences in procedures for knowledge acquisition from maps. Cognitive Psychology, 12, 137-175.

Processing Aspectual Semantics

Sergei Nirenburg

Center for Machine Translation
Carnegie-Mellon University

James Pustejovsky

Computer Science Department
Brandeis University

Abstract

A computational treatment of aspect in English is presented. A set of aspectual values is introduced and discussed. The lexical and contextual clues for determining aspectual values are determined. The structure of the entry in the main dictionary supporting aspectual (as well as other types of) analysis is illustrated. A computational framework for an aspectual analyzer is described, in which the latter is conceived as one of a group of specialist analysis modules working together, in a distributed (blackboard-oriented) computational environment.

1. The Concept of Microtheories

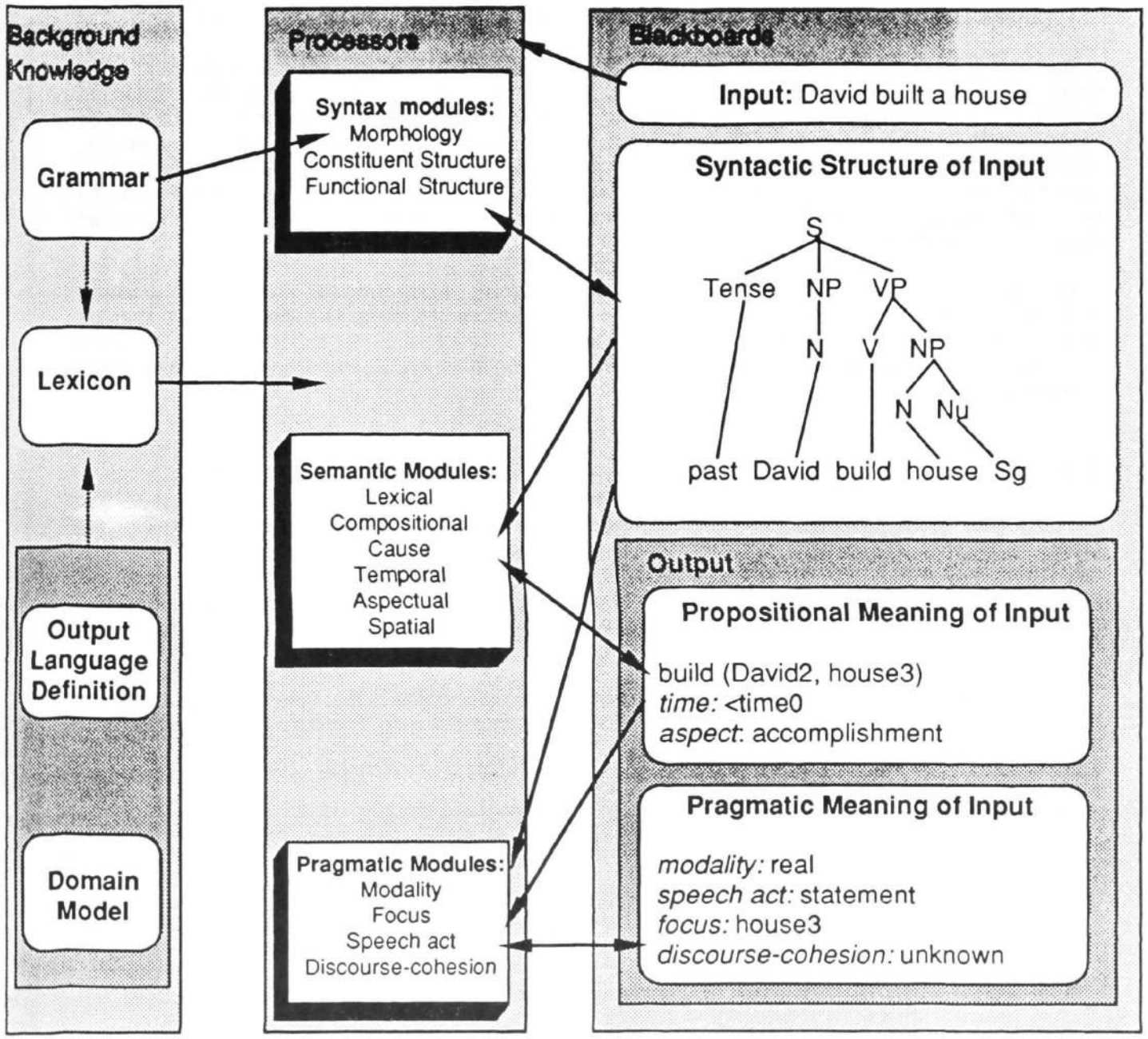
A computational model of language behavior must provide treatment of a large number of syntactic, semantic and pragmatic phenomena. It may be realized in a set of computer programs that obtain natural language inputs, extract their meanings and represent them in a well-defined notation, after which they react accordingly to the message in the input. Of course, some types of reactions may have nothing to do with natural language (for instance, a robot might perform a motoric operation after having understood a verbal command). However, a number of reactions (as in dialog systems or the various text processing systems, such as, for instance, those of machine translation) involves generating natural language texts based on the extracted meanings. Thus, a complete model of language behavior must deal with recognition, representation and synthesis of natural language texts.

Significant progress has been made recently in the field with respect to the theories of syntax. Semantic and pragmatic phenomena have traditionally been less amenable to computational analysis. It does not seem plausible that an integrated semantic theory that covers all of lexical and compositional phenomena as well as the various pragmatic considerations is formulated in the near future. This assessment becomes even more evident if one recognizes the necessity of providing heuristics for automatic recognition of the multiple meaning facets of natural language texts as a part of the theory. At the same time, linguistics has accumulated a significant body of knowledge about the various semantically laden phenomena in the natural languages (cf. Raskin, 1987 for a discussion of how this body of knowledge can be applied to computational analysis).

The above suggests that one of the more feasible ways toward building a comprehensive computational model of language understanding and generation behavior in humans is to develop a large number of *microtheories* that deal with a particular linguistic phenomenon in a particular language or group of languages and then provide a computational architecture that allows the integration of the operation of all the modules based on these microtheories. Thus, one can envisage a microtheory of time, modality, speech act, causality, etc. This paper is devoted to a microtheory of *aspectual meanings*.

To integrate the microtheories we suggest the use of a version of the blackboard computational

architecture, in which a number of processes co-exist and, using a variety of background knowledge modules, collectively produce a desired output. The structure of the language comprehension component of a language behavior model is illustrated in Figure 1. In this figure the processors are computational realizations of the various microtheories derived for the corresponding linguistic phenomena. These processors operate using the data from the background knowledge repositories, such as grammars and dictionaries, as well as the intermediate results stored on the universally accessible set of blackboards. A more detailed description of the model and its components see in Nirenburg and Raskin, 1987a and Nirenburg, 1987.



Background data, knowledge acquisition time → Data, processing time

Figure 1-1: Comprehension Component of a Computational Model of Language Processing

2. Treating Aspectual Meanings: the Task

Aspectual meaning is a component of the overall meaning of a natural language utterance. It is illustrated in examples (1) -- (3) below, in which the only difference between the verb meanings is aspectual: +protracted, -culminative in (1); +protracted, +culminative in (2) and, surprisingly, -protracted, -culminative in (3).

(1) I walked in the park for an hour yesterday
(2) Yesterday I walked to work
(3) At 8 o'clock yesterday morning I was walking to work

Building a semantic analyzer that determines aspectual meanings of input utterances involves specifying a) a set of values for aspectual meanings; b) a set of rules for assigning particular values of aspect; c) the knowledge (the clues) necessary for the formulation of the conditions in these rules (to be found in a) the entries for verbs in the analysis lexicon; b) the syntactic structure of the input utterance; and c) the presence of certain aspectually significant lexical units, such as temporal modifiers or semi-auxiliary 'aspectual' verbs in the input text); and d) an architecture for the analyzer that will allow the results of one processing module (for instance, the syntactic parser) to serve as a decision aid for another module (such as the aspect analyzer).

In what follows we 1) suggest a set of aspectual meaning values, inspired by a theory of aspect described in Pustejovsky (submitted); 2) describe the architecture of a comprehensive semantic analyzer of which an aspect analyzer is a component; 3) describe the structure of the analysis lexicon; 4) survey the language material that is the basis for formulating decision rules; and 5) present a set of analysis rules for aspect.

3. A Language for Aspect-Related Analyzer Output: A Set of Aspect Values

The logical place to attach the information about aspect in a semantic representation is at the propositional level. Propositions are represented as frames that are essentially instantiations of event-types, with their arguments constrained in accordance with evidence in the input text. The arguments in a proposition representation include conceptual case roles, preconditions, effects, temporal, spatial and aspectual values. Within the current architecture, a separate analysis component is assigned the task of determining the contents of each of the above slots.

It is important to understand that our model uses predetermined calculus that determines what different types of events there can be. That is, just as a grammar defines the well-formed syntactic structures in a language, the calculus of aspect contributes to defining what a well-formed semantic structure is. Furthermore, all event-types are built recursively from two primitive event-types, *state* and *process* (see Pustejovsky 1987 for details).

The set of values defined for the aspectual slot in our system is as follows:

A **state** is an event-type which is nondecomposable and does not in itself refer to any initial or endpoint. Note that additional textual information can refer to initial or endpoints of a state or its duration. States for which such additional information is introduced will be called **bounded states**.

A **process** is a possibly ordered set of event-types each of which can be a state or a process itself.

If we consider a process where the initial and/or the final event-types are distinguished so that that single state is on a par in importance with the complement set, taken as a whole, then the resulting event is different. This phenomenon is called *headedness*, and the marked event-types, correspondingly, *heads*.

If the final element in a process is its *head*, and the process itself is semantically well-specified, then

this event-type is called an **accomplishment**. Thus, in an accomplishment there is an embedded process as well as a distinguished final event or state, and both event constituents are of equal status. For example, in *John built a house* the final state of there being a house is as significant as the building process.

Analogous to the above definition, there are some processes in which the final event-type can be distinguished, but the complement set is not semantically specified through the meaning of a lexical item. Such an event-type is called **achievement.** In an achievement the process and its head do not, thus, have equal status, the head being more important.

The initial event of a process can also be distinguished as a head. These are left-headed structures. Left-headed structures arise only when the initial event can be distinguished as causing the rest of the event-types in the ensuing process. The left-headed structures are **causative-processes**.

There are event-types that allow two heads, the left ones and the right ones. If an event-type is double-headed, then the event-types that are neither initial nor final are typically underspecified semantically. The double-headed structures are called **transitions**.

4. The Aspect Analyzer

4.1. An Architecture for Distributed Analysis

The set of routines for determining aspectual values of utterances forms a functional component of a comprehensive natural language analyzer (see Section 1 above). We develop this analyzer, DIANA (Nirenburg, in preparation), as a distributed system of specialist modules working together to produce a complex output structure which is represented in our approach as a set of frames corresponding to the levels of text, sentence, clause, proposition and propositional argument denotations (cf. Nirenburg et al. 1987). Our analyzer shares these properties with the generation system DIOGENES being developed at CMU (Nirenburg, 1987). Together the two systems will form the basis of a knowledge-based machine translation system.

4.2. Knowledge Sources

Figure 1 illustrates the (static and dynamic) knowledge sources necessary to support aspectual analysis in the framework of a distributed, blackboard-based analyzer.

The lexicon, especially the semantico-pragmatic portion of its entries, is a most important knowledge source for determining aspectual values. The structure of our lexicon entries can be described as follows[1].

[1]This is an abridged version of the lexicon entry structure. See Nirenburg and Raskin, 1987b for a detailed discussion

```
AL-entry            ::= ( <SL-pattern> <meaning-pattern>*)
<SL-pattern>        ::= ( SL-Lexical-Unit <lex-info>)
<lex-info>          ::= ( (<syntactic-info>)
                          (morph <inflection-type>))
<syntactic-info>    ::= {the contents of a syntactic dictionary}
<inflection-type>   ::= {an indication of irregularities in forming
                        word forms, e.g., goose - pl. geese}
<meaning-pattern>   ::= ((token-of (value <domain-concept>))
                         (<property> (value <value>*))) |
                         (<property> (value <value>*))*
<domain-concept>    ::= {any concept in the domain model defining a sublanguage}
<property>          ::= {any relation or attribute from the domain model}
<value>             ::= {any concept or attribute (scale)
                         value in the domain model}
```

The lexicon has four kinds of meaning patterns: a) instantiations of concepts in the domain model, e.g., *computer*; b) instantiations of attributes of concepts in the domain model, e.g. *fast*; c) clues for determining the attributes of the properties comprising both the propositional and the pragmatic meaning, e.g., *finish*; and d) clues for making other semantic and pragmatic decisions, such as reference, e.g. *the*.

Group c) includes a class of verbs that have a special significance for aspectual analysis, the so-called 'phase' or 'aspectual' verbs. These verbs have a lexical meaning but no independent ontological meaning. The aspectually relevant meanings that they introduce into the overall meaning of the input can be classified as follows:

- inchoative: *start, begin, resume*
- continuative: *continue, keep*
- completive: *end, finish, complete*
- abortive: *stop, cease*
- iterative: *repeat*

4.3. Material

In this section we analyze a number of characteristic examples with respect to their aspectual values (AVs).

4.3.1. States

(4) John loves Mary
(5) John has loved Mary for 2 years

The aspectual analysis produces: *AV: state; Time: present* for (4) and *AV: bounded state; Time: begin: (NOW - 2 years)* for (5). In (4) the verb *love* is lexically specified as a state[2]. The temporal reference in which the verb is grounded will act to bound this state in any number of ways. In this case, since the tense is the present, there is no delimitation on the state. In (5) however, the present perfect together with the durative adverbial acts to left-bound the state denoted by the proposition.

4.3.2. Processes

(6) John walked yesterday
(7) John walked to work yesterday
(8) John is walking to work

(6) is analyzed as *AV: process; Time: past: yesterday*, (7) as *AV: accomplishment; Time: past:*

[2]that is, listed in the dictionary as being a token of a class of concepts which are descendants of *state* in the hierarchy of concepts that embodies the domain model

yesterday, and (8) as *AV: state S; S is a member of P in accomplishment; Time: present.*

Notice the effect of prepositional modification and that of the progressive on the aspectual value of the sentence. The prepositional phrase in (7) indicates the goal of a process with no intrinsic culmination. This goal acts to the terminate the event and shifts the aspectual type to an accomplishment. In (8), on the other hand, the progressive influences the nature of the resulting aspect-type more than the presence of the prepositional phrase. The result is a a state (see Pustejovsky (1987) for further discussion). The important thing to realize in the interpretation of this sentence is that the goal state is not entailed when the progressive applies to an accomplishment.

4.3.3. Achievements

(9) Bill won a race
(10) Bill is winning the race

(9) obtains the aspectual value of *achievement*. The aspectual class of achievement verbs is probably the most consistent since it seems to resist the modification that leads to aspect-type shifting. Thus, when the verb recognized is lexically specified as achievement, the resulting type will be the same. The one exception to this is the progressive, which has a similar effect as that mentioned above for 'walking to work'. The resulting aspect type in (10) is *state*.

4.3.4. Accomplishments

(11) Fred built a house
(12) Fred built houses for 5 years
(13) Fred was building a house

The analysis for the above brings: *AV: accomplishment* for (11); *AV: process; iterative: element of iteration:* **build a house; AV: accomplishment** for (12); and *AV: state S; Time: past* for (13).

Any accomplishment usually entails a culmination, but in (12) the aspect type is a process. This is a result of the bare plural object, which iterates over the lexical accomplishment to produce a process. Thus, a durative adverbial is permitted. When this sentence is put in the progressive form, it is stative.

4.3.5. Causative-Processes

(14) Max sent a package
(15) Max sent a package to Leo

(14) is analyzed as *AV: causative-process*, (15) as *AV: accomplishment.*

Here the process is directed by a single initiating event (the projecting), and thereafter there is no agency involved. Hence, in (15) it is not entailed that Leo *receives* the package (cf. Pustejovsky (1987)).

4.3.6. Transition

(16) John gave the book to Mary
(17) John gave books to Mary

The analyses: *AV: transition* for (16) and *AV: process, iterative; element: transition* for (17).

The most noticeable thing about these examples is the absence of any specified mode of transition. That is, what is lexically specified in the semantics of *give* is simply the beginning and end states of an event. Notice that in (17) the bare plural object shifts the interpretation to a process, as with the accomplishments.

4.4. The Analysis Rules for Aspect

The above linguistic material shows that many types of knowledge come into play in determining the aspectual value of a clause: lexical-semantic, syntactic and contextual. Specifically, this information is drawn from

- entries for verbs in the analysis lexicon
- verb tense values from the results of the syntactic module of the analyzer
- the presence of particular modifiers (adverbs and prepositional phrases) in the input (the syntactic module being responsible for determining what is modified by what)
- the presence of aspectual verbs in the input in syntactically relevant positions
- the presence, the meaning and the syntactic form of case-role holders for particular verbs (the above are determined through the operation of the syntactic and the propositional-semantic modules of the analyzer)

Specific rules and heuristics that the analysis system uses to identify the aspectual type for an input sentence include the following[3]:

- If the main predicate in an input sentence has the aspectual marker *achievement* in the dictionary, assign the aspect value *achievement*
- If the main predicate is marked in the dictionary as *state* **or** if the morphologo-syntactic analysis determines that it is in a progressive form, assign the aspect value *state*
- If the main predicate is marked as *accomplishment* **and** the direct object in the sentence is definite **or** the main predicate is marked as *process* **or** *directed-process* **and** is modified by a prepositional phrase, assign the aspect value *accomplishment*
- If the main predicate is marked as *process* **or** *directed-process* **or** a direct object is present identified as *bare-plural* **or** *mass-noun*, assign the aspect value *process*.
- If an aspectual verb is present in the sentence follow these rules:
-
    - Inchoative + (State **or** Process **or** accomplishment) = achievement; other combinations are impossible for inchoative
    - Continuative applies only to a process to yield a process
    - Completive and abortive apply to a process to yield an achievement
    - Iterative applies to achievements, accomplishments and transitions to yield processes.

5. Discussion: Future Work and Limitations

Passonneau (1987) and Moens and Steedman (1987) are two of the recent publications devoted to a similar topic.

The problem specifications in this model and Passonneau (1987) are very compatible. Still, the representation of the aspect calculus assumed here is richer than Mourelatos' (1981) typology, which Passonneau assumes, in that it proposes a larger number of primitive event-types, which, we believe provide a more accurate coverage for the aspectual phenomena. Space does not permit us to adequately motivate all our categories, but see Pustejovsky (1987) for discussion. Similarly, the set of analysis clues we employ is broader, including the input from lexical sources beyond the verb itself. Differences in

[3]We decided to present these heuristics not in the format used by our program but rather in a human-readable form

analyzer architecture are also quite significant.

Moens and Steedman (1987) pursue a somewhat different goal. They emphasize how aspectual processes fit into a larger temporal reasoning component to provide a richer tense system. This is also explored in Pustejovsky and Herman (1988). But our concerns in this paper are primarily elsewhere. We have addressed the problem of the lexical specification of a verbal element and how this value changes in the context of other propositional information.

The aspectual analyzer, as described above, has been prototyped. It can be tested in earnest only when at least several other modules of DIANA are implemented. As to the enhancements to the aspect module itself, we plan to extend the coverage of the aspect determination heuristics through empirical studies over large text corpora.

Bibliography

Corkill, D.D., K.Q. Gallagher, and K.E. Murray. 1986. GBB: A Generic Blackboard Development System. AAAI-86. 1008-1014.

Moens M. and M. Steedman. 1987. Temporal Ontology in Natural Language. ACL-87.

Mourelatos, A.P.P. 1981. Events, Processes, and States. In: P. Tedeschi and A. Zaenen. Syntax and Semantics, vol. 14. Tense And Aspect. NY: Academic Press.

Nirenburg, S. 1987. A Distributed System for Language Generation. Technical Report CMU-CMT-86-102. Carnegie-Mellon University. May.

Nirenburg, S., in preparation. DIANA: A Distributed Semantic Analyzer.

Nirenburg, S., V. Raskin and A. Tucker, 1986. On Knowledge-Based Machine Translation. Proceedings of COLING-86. Bonn, pp. 627-632.

Nirenburg, S. and V. Raskin, 1987a. The Subworld Concept Lexicon and the Lexicon Management System. *Computational Linguistics*, vol. 13, 1987, issue 3-4. (with V. Raskin).

Nirenburg, S. and V. Raskin. 1987b. The Analysis Lexicon and the Lexicon Management System. *Computers and Translation*, 2, pp. 177-188.

Passonneau, R. 1987. Situations and Intervals. ACL-87.

Pustejovsky, J. 1987 An Event Stucture for Lexical Semantics. Submitted to *Computational Linguistics*.

Pustejovsky, J. and I. Herman, 1988, Subevent Reasoning and Temporal Logic. Submission to *AAAI-88*.

Raskin, V. 1987. Linguistics and Natural Language Processing. In: S. Nirenburg (ed.). **Machine Translation: Theoretical and Methodological Issues**. Cambridge University Press, pp. 42-58.

Gain Variation in Recurrent Error Propagation Networks

Steven J. Nowlan
Department of Computer Science
University Of Toronto[1]

INTRODUCTION

Neural networks have received much attention recently as plausible models for studying the computational properties of massively parallel systems. Learning algorithms have been developed (Rumelhart et al., 1986; Ackley et al., 1985) that enable these networks to learn internal representations, allowing them to represent complex non-linear mappings. Two distinct types of networks have been studied quite extensively. The first of these uses analog (continuous-valued) units with a sigmoidal I/O function (Hopfield and Tank, 1985), and an error propagation algorithm for updating the weights to minimize an error function (Plaut et al., 1986; Plaut and Hinton, 1987). Most of these studies have focused on strictly feedforward networks. The second type of network employs stochastic binary units and symmetric connections. From an initial state these networks approach a low temperature fixpoint (stable state), which represents a local minimum of a global energy function. The weights in such networks may be updated by examining the difference in statistics between the states with inputs clamped and unclamped at thermal equilibrium (Ackley et al., 1985).

Recent work has shown some interesting relationships between these two distinct models. Peterson and Anderson (1987b) have developed a continuous approximation to the Boltzmann machine algorithm, in which the stochastic binary units of the Boltzmann machine are replaced with analog units whose states are mean field approximations to the average states of corresponding stochastic binary units at equilibrium. They have shown significant speedup in convergence and improved generalization for interesting problems (Peterson and Anderson, 1987a). Hopfield (1987) has shown that for a certain class of statistical estimation problems, the statistical network and the analog network have very closely related properties and learning algorithms. Provided that four conditions are met, the error propagation update rule for the weights in an analog feedforward network is a mean field approximation to the update rule for a statistical network using the Boltzmann machine algorithm. These four conditions are: the analog network must use a symmetric divergence (Pearlmutter and Hinton, 1986) rather than the more common mean square error function; the statistical averaging in the two state network is performed over the hidden and output units, but not the input units; the two networks have a small number of outputs; and the networks have only a single layer of non-interacting hidden units[2].

This work explores another parallel between statistical and analog networks. Recurrent analog networks often show better convergence if a global gain term is introduced which may be varied over a single settling (Hopfield, 1984). The result is a procedure similar to simulated annealing (Kirkpatrick et al., 1983). An error propagation scheme is presented which allows an analog network to "learn" its own gain variation schedule, and experimental results for a constraint satisfaction task show an order of magnitude speedup in learning when this approach is used.

ERROR PROPAGATION IN RECURRENT ANALOG NETWORKS

The recurrent analog networks that are considered here use a synchronous updating procedure, and place no restrictions on the nature of the connectivity matrix. The activation level of a unit at time t is a non-linear function of the input to the unit at time t:

$$y_{j,t} = g(x_{j,t}) \tag{1}$$

One possibility for g, used in our simulations, is the logistic function, $g(x) = 1/(1+e^{-x})$. The input is a weighted sum of the states of units in the previous time step:

$$x_{j,t} = G_t \sum_i w_{ji} y_{i,t-1} \tag{2}$$

The term G_t is a global gain term which premultiplies the input for every unit. There are also two distinguished subsets of units: $\mathcal{I}$ the set of input units and $\mathcal{O}$ the set of output units. The states of units in $\mathcal{I}$ are determined by the environment.

[1] The author is currently visiting the University of Toronto, while completing a Ph.D. at Carnegie-Mellon University

[2] This means that to first order the hidden units have no effect on each other.

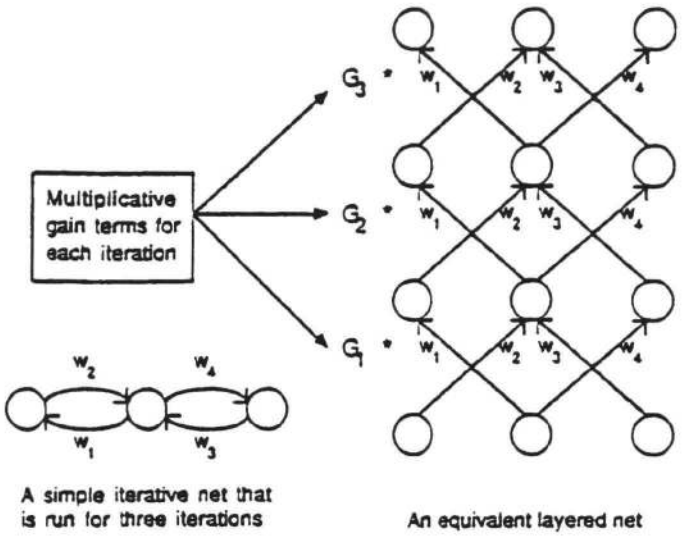


Figure 1: A recurrent network and the equivalent layered network. Corresponding weights in layers must be identical (*i.e.* w_1 has the same value in all layers), but the gain terms (G_i) vary between layers. Each layer corresponds to the state of the recurrent network at a different point in time.

The state vector for the recurrent network at time t can be treated as the vector of states of the tth layer in an equivalent layered network (figure 1). The trajectory of the state vector for the recurrent network is then represented by the state vectors of successive layers in the equivalent network. Since the weights in the recurrent network are stationary (fixed during the settling of the network), the set of weights between successive layers in the layered representation must be identical.

The dynamics for these networks can be expressed by finding the continuous differential equations equivalent to the discrete difference equations (1) and (2). This produces the following set of coupled differential equations:

$$\frac{\mathrm{d}y_j}{\mathrm{d}t} = -y_j + g(x_j) + I_j \tag{3}$$

where

$$x_j = G \sum_i w_{ji} y_i \tag{4}$$

I_j is defined to be zero for all units not in $\mathcal{I}$. Equation (3) is a simple transformation of an equation that has been studied by both Amari (1972) and Hopfield (1984). Amari showed that in randomly connected networks with the dynamics of (3) the attractors were either stable or bistable.

When the attractors are fixpoints, one can use such networks to perform constraint satisfaction searches. The activities of units encode the values of the parameters of a problem, and the weights on the connections encode the constraints between the parameters. This approach has been used for classic optimization problems (Hopfield and Tank, 1985) and for parsing (Selman and Hirst, 1987). Coding the constraints into the weights by hand becomes a formidable task for large problems. This raises the possibility of devising learning algorithms which will manipulate the weights of a recurrent network to model the constraints of a specific problem through some training procedure.

There are several ways in which to pose the problem of modifying the weights. One approach would be to consider the fixpoint of the network for a specific input, and compute some error measure based on the distance of this fixpoint from a desired fixpoint. One could then use gradient descent to modify the weights to minimize this error measure (Pineda, 1987). An alternative is to consider not just the fixpoint, but the entire trajectory of the network. This approach was first suggested by Rumelhart and Hinton (1986) and is the approach taken here.

Consider a trajectory of length k for a recurrent net-

work. There is an equivalent layered representation of this trajectory which has k distinct layers (figure 1). The standard back-propagation algorithm may be applied to this layered representation, if we define an error measure E for the final states of units in $\mathcal{O}$. Using this approach we can derive partials for the weights:

$$\frac{\partial E}{\partial w_{ji}} = \sum_t \frac{\partial E}{\partial x_{j,t}} G_t y_{i,t-1} \tag{5}$$

Note that the back-propagation proceeds through the sequence of states in the trajectory, and in particular that the partial of E with respect to w_{ji} will vary along that trajectory. Since the weights are stationary, some form of time averaging is required. Equation (5) uses a uniform time averaging, although versions which favour the terms near the end of the trajectory could also be used.[3] Thus the weights are being updated based on the average derivative over the trajectory. The disadvantage of using this approach to modifying the weights in the network is that it becomes necessary to store the entire trajectory for the back propagation phase. However, the advantage is that the network can be trained not just to have certain limit behaviour, but also to have certain behaviour along the trajectory followed to the limit. One obvious example is to force the network to learn fixpoints as attractors, by penalizing bistable behaviour during the last few states of the trajectory. To allow control over the trajectory, the back propagation procedure is modified slightly to allow directly observed error terms in time steps other than the last to be added to the back propagated terms for units in $\mathcal{O}$. (This is equivalent to specifying desired states for intermediate layers in a layered network.)

The term G_t in equation (5) determines the steepness of the non-linearity in the recurrent network, and has been referred to as the system gain (Hopfield and Tank, 1985). This term may be allowed to vary during the settling of the network. For example, an increasing gain in a network with mutually inhibitory connections can implement a winner-take-all network that converges quickly. Hopfield and Tank (Hopfield and Tank, 1985) found that increasing the gain slowly as their analog network settled increased the quality of the solution found by the network. They suggested that increasing the gain in this fashion was analogous to following the effective field solution from a high temperature, resulting in a final state near the thermodynamic ground state of the system.

Rather than determining a gain schedule in advance, an error propagation scheme can be used to decide how G_t should vary over the trajectory of the network. G_t is optimized by performing gradient descent in the error measure:

$$\frac{\partial E}{\partial G_t} = \sum_j \sum_i \frac{\partial E}{\partial x_{j,t}} w_{ji} y_{i,t-1} \tag{6}$$

Once again averaging is necessary, in this case over all units in the network, since G_t is a global premultiplier.

Equations (5) and (6) can be combined to produce an algorithm that will allow networks to exhibit desired trajectories from initial states. These trajectories correspond to a constraint satisfaction search via relaxation. An algorithm of this form is described by Nowlan (Plaut et al., 1986).

PERFORMANCE ON A CONSTRAINT SATISFACTION PROBLEM

The author investigated the performance of the gain variation error propagation algorithm through simulations on some small problems (Plaut et al., 1986). For simple coding and sequencing tasks a network that learns a variable gain schedule learns to solve a problem several times faster than a similar network with fixed gain. In addition, the learned gain variation schedules outperformed several hand designed schedules on the same tasks. These tasks are all expressed in terms of I/O mappings, a prespecified output was required for each input. This makes the dynamics to be learned quite a bit easier. It is possible for the network to learn a trajectory from each input to the desired output without constructing a true attractor (a stable state with low error and a region of attraction around it) for that output.[4] A task in which it is necessary to construct robust attractors which represent the problem constraints provides a much richer domain in which to study the performance of gain variation.

The problem selected is the *n queens* problem. This is a classical constraint satisfaction problem that was studied extensively by early AI researchers (Nilsson,

[3] The G_t terms actually do weight the derivatives, so the time averaging is not truly uniform.

[4] Additional simulations showed that this was in fact the case for several of the experiments discussed.

1980; Feigenbaum and Barr, 1981). The general problem is to place n queens on an n by n grid of squares, such that there is no vertical, horizontal, or diagonal line through the grid that contains more than 1 queen. This problem is easily mapped into a network: Each cell of the grid is represented by a unit in the network, and each unit is fully connected to every other unit including itself.[5] In addition, each unit has an external input line to carry environmental input. In this special case every unit is both an input and an output unit ($\mathcal{I} = \mathcal{O}$). One nice feature of this problem is that it can be easily scaled.

Given a random initial state vector, the network is required to settle into a final state which represents a valid solution to the n queens problem. A valid solution is one in which n units are above the *on level* (0.9), and all the rest are below the *off level* (0.1). In addition, no two *on* units can lie on the same vertical, horizontal, or diagonal line. To solve this problem, the network must construct stable attractors for the valid solutions to the problem and the set of attraction basins must span the entire input space.

Since this task differs from the typical I/O mapping tasks given to error propagation algorithms, some care must be taken in deciding on an error measure. Given a random initial input state, there are in general many final states which are equally acceptable as solutions. One possibility is to measure the distance of the actual output from each of these final solutions, and take the minimum distance as the error to be propagated. This results in a form of nearest neighbour error measure.

An even more sophisticated training method, a form of shaping, may be used to produce good results and reduce the training time. The network is first presented with noisy versions of solution vectors as input. A solution is chosen at random, and then noise uniformly distributed between 0 and η (initially 0.2) is added to units that are off and subtracted from units that are on. This noisy vector is clamped to the external inputs, and the network allowed to settle for γ cycles. During the last three cycles, the mean square distance between the output vector and the solution vector used to generate the input is calculated as the error measure. The error is taken over the last three cycles to force the network to learn attractors which are fixpoints. The network is trained in this fashion until its average error (normalized by the vector lengths) is less than 20 percent over the last 50 trials. At this point, the same training regime is continued, except that the input is presented during the first cycle only, rather than being clamped. During this second phase η is gradually increased to 0.4. Once the average error over 50 trials is less than 10 percent, a final training phase is performed in which initial states are randomly generated, and the nearest neighbour error measure is used over the last three trials. The first phase of training establishs the attractors, the second phase stabilizes the attractors independant of external input, and the third phase ensures robustness of the attractors.

The simulation results are summarized in Table 1. One obvious anomaly in the table is the difference between the 5 queens and 6 queens problems. The gain variation technique shows a clear advantage for the 5 queens problem (an order of magnitude improvement), but the performance with and without gain variation is nearly identical for the 6 queens problem. The answer lies in the second column of the table. There are only 4 solutions for the 6 queens problem, and two of these solutions are simple mirror images of the other two. It is quite easy for the network to set the unit biases to favour the small set of units which appear in these 4 solutions. This "trick" makes establishing stable attractors quite easy. On the other hand for the 5 queens problem, and the larger problems, there are sufficiently many solutions so that almost every unit is active in at least one solution, so the biases alone cannot be used to give the network a head start. Under these circumstances, the dynamical behaviour of the network becomes much more important, and so a much stronger advantage is shown by the gain variation algorithm.

The set of weights learned by the network for one example of the 5 queens problem (figure 2) shows that in its solution the network has extracted the essential constraints of the task. Each unit has learned to develop a positive weight to itself, and negative weights along all horizontal, vertical and diagonal lines which the unit lies on. This means that all of the units have a natural tendency to turn on, and along each line (in any orientation) a winner-take-all network has formed, so that the stable states are those in which only one unit on a particular line is on. This is a very natural representation of the original problem constraints, which stipulated that no

[5] The network must learn which of these connections are really needed to solve the task.

		No Gain Variation				Gain Variation			
n	Solutions	Phase 1	Phase 2	Total	ϵ	Phase 1	Phase 2	Total	γ-ϵ
4	2	115	420	677	0.1	70	232	350	1.0×10^{-3}
5	10	6371	11048	19820	0.01	371	1048	1540	1.0×10^{-4}
6	4	371	1048	1420	0.1	397	698	1242	1.0×10^{-3}
7	40	20600	57480	83200	0.01	1280	4160	6080	1.0×10^{-4}
8	92	50000	—	50000	0.01	1960	12600	16200	1.0×10^{-4}

Table 1: Simulation results for various sizes of the n queens problem. ϵ is the size of the weight step, γ-ϵ is the size of the gain update. The numbers under the columns Phase 1, Phase 2, and Total are the number of weight updates performed in each training phase (see text) and in total. One update was performed after every 20 training examples. The results for the 4 queens problem are averaged over 50 runs, for the 5 and 6 queens problems the averages are over 20 runs. Only 2 runs are reported for the 7 queens, and 1 run for the 8 queens. The algorithm with no gain variation was not able to meet the 20 percent error criteria for phase 1 for the 8 queens problem and was terminated after 50000 updates.

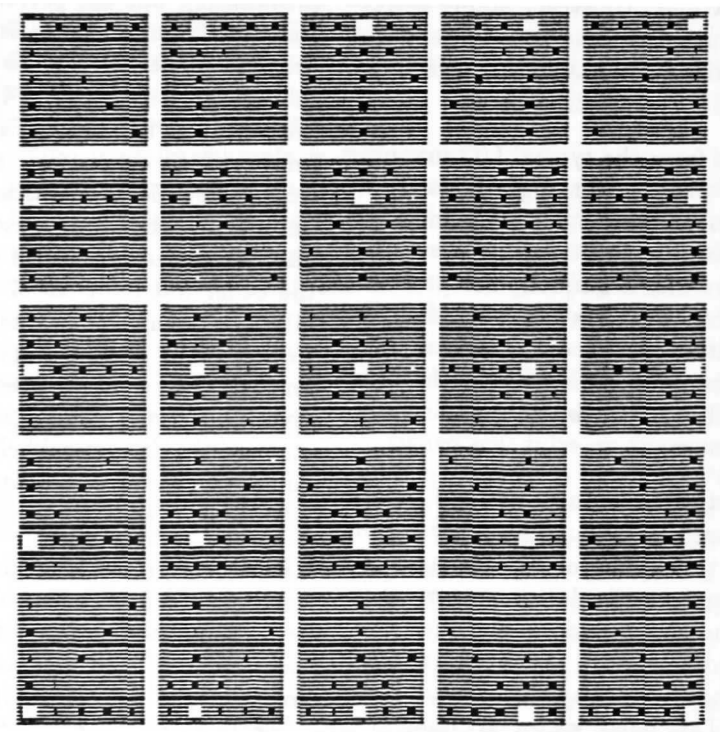


Figure 2: Weights learned for the 5 queens problem. The weight display is recursive, each large grey square represents one unit. Within each large grey square is a 5 by 5 grid of squares which represent the weight of the connection from that unit to every other unit in the network. Black squares represent negative weights, white squares represent positive weights, and the size of the square represents the magnitude of the weight. Weight decay was applied to drive all non-essential weights to zero, to simplify the weight display.

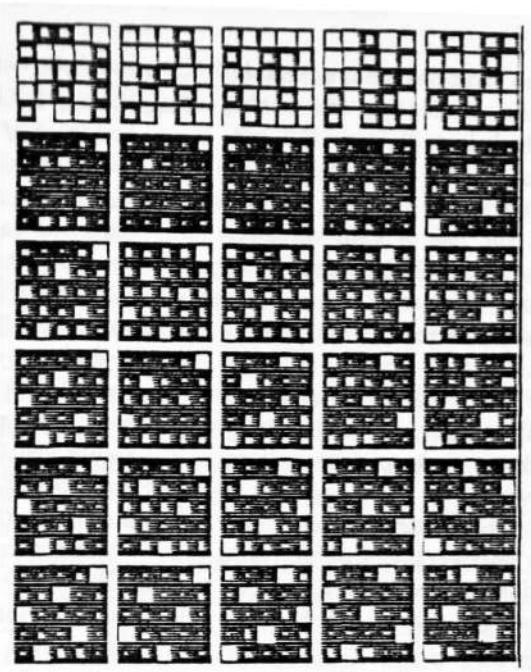


Figure 3: Display of the dynamics of the 5 queens problem for 5 different initial states. Each column represents the settling of the network for one case. The large black squares contain the activity levels of all 25 units in the network, arranged in a 5 by 5 grid. The size of the white square in each grid position is proportional to the activity of the corresponding unit. In all cases the state reached at the end of 6 cycles is a stable fixpoint.

two queens could lie on the same horizontal, vertical or diagonal line.

The dynamic behaviour of a 5 queens network can be seen in figure 3. In the third of the 5 examples we can see the network placing a queen in the third row when the initial configuration contains no dominant unit in this row. In the fifth example we can see competition between two initially dominant units in both the first and second rows, and also the creation of a dominant unit in the third row. The performance of the network is quite robust, even on highly ambiguous inputs.

Our experiments with the simple I/O mapping networks showed a tendency for the learned gain schedules to be "annealing" schedules, starting at a low gain and increasing it as the network settled (Plaut et al., 1986). This same effect is observed in the simulations for the n queens problem.

DISCUSSION

Varying the gain for a recurrent network as it settles has been suggested elsewhere (Hopfield and Tank, 1985; Pineda, 1987), as has an analogy between gain in recurrent analog networks, and temperature in statistical networks (Amari, 1972; Hopfield, 1984; Pineda, 1987). The unique aspect of this work is the use of an error propagation scheme to *simultaneously* optimize the weights and gain schedule for a recurrent network. The empirical results presented here show that for constraint satisfaction problems in which a complex attractor structure must be developed, the parallel optimization of the weights and gain schedule can produce an order of magnitude speed-up in the convergence of the optimization. What is not clear is whether this speed-up is obtained by following a shorter path to the same weight region that would be reached by optimizing the weights alone, or whether a qualitatively different region of weight space is reached by the combined optimization.

Some simple simulations tend to support the latter hypothesis. If an n queens network that has been trained with gain variation has its gain schedule modified so that the gain is a constant value (for example the mean of the gain schedule), qualitatively different behaviour is observed from the network. Additional stable states are observed, which do not correspond to solutions to the n queens problem, and which are not stable when the network is

allowed to settle using the original gain schedule. This behaviour is expected. Annealing the analog network with the gain schedule will force the network into states nearer the global error minimum, and away from local minima with relatively high error. It would appear that the use of gain variation allows the network to find a set of weights that has an attractor structure with many high error spurious attractors, in addition to the low error attractors that correspond to solutions to the task. It is apparently much easier to find a set of weights with an attractor structure of this form, rather than one with attractors which correspond only to task solution points. This leads to the hypothesis that the speed-up in convergence is obtained by allowing a much larger region of weight space to satisfy the problem, and that the combined optimization leads to a qualitatively different region of weight space than by optimization of the weights alone.

There is an additional factor which may account for some of the speed-up in learning observed with the parallel optimization of the weights and gain schedule.[6] The error surface for many problems that back propagation is applied to is characterized by ravines with steep sides in most directions, but a shallow descent in one direction. Once the weight vector is aligned with the floor of the ravine, one can move quite rapidly along the floor of the ravine by simply adjusting a global gain term. To examine this effect, several of the 5-queens simulations were repeated with a gain term that was learned by error propagation, but was constrained to be constant during a settling (as the weights were constrained). This algorithm was a factor of two to three faster than the simple back-propagation algorithm, but still five to six times slower than the parallel optimization of the weights and gain schedule, suggesting that the ability to scale all of the weights accounted for a small part of the speed-up observed in the n-queens problem. Nevertheless, the speed-up was significant enough to suggest using a similar scale factor in layered networks.

REFERENCES

Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, 9:147–169.

Amari, S. (1972). Characteristics of random nets of analog neurons. *IEEE Transactions on Systems, Man, and Cybernetics*, 2(5):643–657.

Feigenbaum, E. A. and Barr, A., editors (1981). *The Handbook of Artificial Intelligence*. Volume 1, Pitman, London.

Hopfield, J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Nat. Acad. Sci. USA, Bio.*, 81:3088–3092.

Hopfield, J. and Tank, D. (1985). "neural" computation of decisions in optimization problems. *Biological Cybernetics*, 52:141–152.

Hopfield, J. J. (1987). Learning algorithms and probability distributions in feed-forward and feed-back networks. *PNAS*, 84:8429 - 8433.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220:671–680.

Nilsson, N. J. (1980). *Principles of Artificial Intelligence*. Tioga, Palo Alto, CA.

Pearlmutter, B. A. and Hinton, G. E. (1986). G-maximization: an unsupervised learning procedure for discovering regularities. In Denker, J., editor, *Neural Networks for Computing*, American Institute of Physics.

Peterson, C. and Anderson, J. (1987a). *Neural Networks and NP-complete Optimization Problems: A Performance Study on the Graph Bisection Problem*. Technical Report MCC-EI-287-87, Microelectronics and Computer Technology Corportation, Austin, TX.

Peterson, C. and Anderson, J. (1987b). *A Mean Field Theory Learning Algorithm For Neural Networks*. MCC Technical Report EI-259-87, Microelectronics and Computer Technology Corporation, Austin, TX 78759-6509.

Pineda, F. (1987). Generalization of back propagation to recurrent and higher order neural networks. In *Proceedings of IEEE Conference on Neural Information Processing Systems*, IEEE, Denver, Colorado.

Plaut, D., Nowlan, S., and Hinton, G. (1986). *Experiments on Learning by Back Propagation*. Department of Computer Science CMU-CS-86-126, Carnegie-Mellon University, Pittsburgh,PA.

Plaut, D. C. and Hinton, G. E. (1987). Learning sets of filters using back-propagation. *Computer Speech and Language*.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning internal representations by back-propagating errors. *Nature*, 323:533–536.

Selman, B. and Hirst, G. (1987). Parsing as an energy minimization problem. In Davis, L., editor, *Genetic Algorithms and Simulated Annealing*, pages 155–168, Pitman, London.

[6] This was suggested by Geoff Hinton and Mike Mozer, personal communication.

CONJOINT SYNTACTIC AND SEMANTIC CONTEXT EFFECTS: TASKS AND REPRESENTATIONS

Padraig O'Seaghdha
University of Rochester

Abstract

Syntactic and semantic relatedness were orthogonally varied in a series of experiments by presenting semantically related and unrelated noun and verb targets in phrasal contexts syntactically disposing to nouns or verbs. In addition, the subjects' task, naming or lexical decision on the target, was varied across experiments. In lexical decision, semantic facilitation and inhibition effects depended on context-target match, especially for noun targets. In several experiments, naming data showed only weak semantic effects, which were not modulated by context-target match. However, there was clear evidence of syntactic inhibition in these experiments. Finally, robust semantic facilitation was observed in a naming experiment where contexts and targets were always syntactically matched. Thus, although in some experiments lexical decision appeared to reflect additional text-level integration processes to which naming was immune, the naming task was less consistent across experiments. This contradiction may be resolved if a distinction is introduced between situations where lexical targets are part of the sequence being tested and situations where they are external probes.

When a sentence-final content word is presented in a speeded response task, it may vary in semantic and syntactic congruity with the context. In the case of syntactic congruity, several recent studies show that, when a sentence-final target is syntactically anomalous, latencies are slowed (Wright & Garrett, 1984; West & Stanovich, 1986). But most of the research in this area has been concerned with semantic effects, assessed on a goodness-of-completion continuum (e.g., Fischler & Bloom, 1979; Stanovich & West, 1983). In recent work (O'Seaghdha, 1986, in press) I have shown semantic effects in simple phrases where relatedness is defined by a single contextual content word (see Table 1). Just as with more complex contexts, lexical decision responses to related targets were fast relative to the neutral condition, and responses to unrelated targets were slow. However, the effect of the context was virtually eliminated when the function words between the contextual content word and the target were incoherent (e.g. with tea the and COFFEE). This result argues against an important role of lexically mediated facilitation in natural language processing. That is, the mere occurrence of a lexical relative of the target is not sufficient to produce an effect. Rather, it appears that a sequence must be syntactically coherent up to the point of target presentation.

In the experiments reported here, this work was extended by marking both syntactic and semantic appropriateness on phrase-terminal targets (see Table 1). In these experiments, related or unrelated noun or verb targets were presented in noun contexts (contexts in which noun but not verb completions were grammatical), or in the reciprocal verb contexts. The targets were also tested in neutral contexts which lacked close lexical relatives. The sequences were always grammatical up to the

TABLE 1
Examples of Materials Used in the Experiments

Context Words					Related/Unrelated Target			
1	2[1]		3	4	Noun		Verb	
Noun context								
with	tea	/dessert	or	with	COFFEE/	BONE	STIR/	RELAX
Verb context								
the	tea	/dessert	that	you	COFFEE/	BONE	STIR/	RELAX

1 The related(e.g. tea) or neutral(e.g. dessert) context is defined in Position 2. Nominally related targets were always presented in the neutral contexts.

point of target presentation. This procedure provides a stronger test of lexical mediation than the earlier work where the sequences were asyntactic before the targets were presented. In addition, it allows for examination of the conjoint effects of syntactic coherence and semantic relatedness. At least three processes could influence performance under these conditions:

1) Binding of syntactic categories to positional slots. In each case, a strong preference exists for a noun or verb ending. Presenting a member of the wrong syntactic category violates syntax and should slow latencies.

2) Semantic facilitation independent of syntactic binding. This could be mediated by intralexical facilitation (Forster, 1979) or by conceptual-lexical priming (Tanenhaus, Dell, & Carlson, 1987). However, if sentence-context effects depend on syntactic coherence (O'Seaghdha, in press) these effects should be weak.

3) Processes operating after syntactic binding of a target. Such processes would be conditional on the syntactic appropriateness of the target and would reflect success or failure at integrating new words with higher-level contextual representations. Such effects are likely to influence a binary decision task more than the univocal naming task (see Seidenberg, Waters, Sanders, & Langer, 1984; Lucas, Tanenhaus, & Carlson, 1987).

Method

The general procedure in all of the experiments was the same. Eighty related noun-noun pairs used in previous research (O'Seaghdha, 1986, in press) were embedded in simple noun-expectancy phrases (see examples in Table 1). In the neutral condition, a noun unrelated to the target was substituted in each context. To produce the unrelated conditions, noun targets were reassigned to unrelated contexts. Next, a related verb was found for each context, and unrelated verb conditions were generated in the same way as for the nouns. Finally, a set of verb-expectancy phrases was written to provide a symmetrical set of verb context conditions.

Thus, on a particular trial, a subject could read a priming or neutral, noun or verb context. In priming contexts, a related or unrelated, noun or verb target was presented. In neutral contexts, the target was by definition not stongly related, but the corresponding nominally related noun or verb was presented.

On each trial, the four context words were presented serially at a fixed 400 msec rate at a center screen location. In the lexical decision experiments, there were eighty word and forty nonword targets, balanced over conditions. In the naming experiments, the conditions were interleaved in such a way that 120 word targets could be presented without repetition of items.

Summary of Experiments

Table 2 summarises data from seven naming and lexical decision experiments. The conditions map onto experiments, indexed by numbers in the table, as follows:
Experiments 1 and 2 (Lexical Decision, Separate): Noun and verb targets were presented in noun contexts (Experiment 1) and verb contexts (Experiment 2).
Experiment 3 (Lexical Decision, Matched): Contexts and targets were matched: Noun targets were presented in noun contexts. The data are extracted from a larger experiment (O'Seaghdha, in press, Experiment 5).
Experiment 4 (Naming, Mixed): Noun and verb targets were presented in noun and verb contexts.
Experiments 5 and 6 (Naming, Separate): Same as Experiments 1 and 2 except for task.
Experiment 7 (Naming, matched): Nouns in noun contexts, verbs in verb contexts within the experiment.

The Table shows data from all of these experiments organised by type of context and type of target. Sixteen subjects served in Experiment 3, twenty-eight subjects in Experiments 6 and 7, and thirty-two subjects in each of Experiments 1, 2, 4, and 5. The overall relatedness effect is the difference between unrelated and related conditions. The facilitation effect is the difference between neutral and related contexts.

Results

Relatedness The main effect of Relatedness was significant in all experiments except Experiment 4 (Naming, Mixed contexts). Experiment 1 replicates the strong syntactic dependency observed in the previous lexical decision research (O'Seaghdha, 1986, in press). This conclusion holds whether the priming of nouns in noun contexts is compared to the 2 msec priming of verbs in the same contexts within the experiment, or to the 14 msec priming of the noun targets in the verb contexts of Experiment 2. However, the relatedness effects in Experiment 2 are weaker than in Experiment 1 and do not show syntactic dependency. It appears that the verb targets are less strongly related to the contexts than the nouns, so that in Experiment 2 the modest semantic effect of asyntactic nouns is comparable to the modest integration observed for the syntactically congruent but semantically weak verbs.

TABLE 2
Means of Response Latencies with Estimates of Overall Relatedness and Facilitation Summarised from Seven Experiments

Task and Contexts	Target	R	U	N	Relatedness	Facilitation[1]
Noun Contexts						
Lexical Decision, Separate (1)[2]	Noun	498	541	527	43	29
	Verb	590	592	598	2	8
Lexical Decision, Matched (3)	Noun	491	530	519	39	28
	Verb	-	-	-	-	-
Naming, Mixed (4)	Noun	471	479	474	8	3
	Verb	497	504	508	11	7
Naming, Separate (5)	Noun	450	458	456	8	6
	Verb	468	480	480	12	12
Naming, Matched (7)	Noun	450	469	467	19	17
	Verb	-	-	-	-	-
Verb Contexts						
Lexical Decision, Separate (2)	Noun	512	526	524	14	12
	Verb	525	543	532	18	11
Lexical Decision, Matched	No data					
Naming, Mixed (4)	Noun	491	498	501	7	10
	Verb	493	497	491	4	-2
Naming, Separate (6)	Noun	471	481	476	10	5
	Verb	466	481	476	15	10
Naming, Matched (7)	Noun	-	-	-	-	-
	Verb	469	487	480	18	11

1 R = Related; U = Unrelated; N = Neutral; Relatedness = (U - R); Facilitation = (N - R).
2 Numbers in parentheses index Experiments (see text).

The virtual absence of any relatedness effect in Experiment 4 was unexpected. The naming task is considered to be a relatively direct measure of context effects (e.g., Seidenberg et al., 1984). Effects of the kind usually observed in sentence-context studies (e.g., Stanovich & West, 1983) were therefore expected. That is, the effects should resemble those with lexical decision though they might be a little smaller, especially on the inhibitory side. In Experiments 5 and 6, when the contexts were either all noun or all verb, the relatedness effects were stronger, but did not discriminate between syntactically

congruent and syntactically incongruent conditions. This outcome makes sense for Experiment 6 which compares directly to Experiment 2. However, a stronger effect of nouns in noun contexts was expected in Experiment 5. Only in Experiment 7, where contexts and targets were always matched, was the expected large facilitation effect for congruent nouns observed.

Syntactic inhibition Although semantic effects were virtually absent in Experiment 4, robust syntactic inhibition was observed, at least for noun contexts. This result is important for the practical reason that it provides assurance that subjects did not merely fail to register the contexts in this multi-condition experiment, and for the substantive reason that it shows an independence of semantic and syntactic effects. Independence of semantic relatedness and syntactic congruity was also observed in Experiments 2, 5 and 6. In the case of Experiments 2 and 6, this may be attributed to semantic effects for nouns balancing syntactic integration for verbs. But in Experiment 5, the same discrimination between nouns and verbs that was observed in Experiment 1 was expected.

Nouns and verbs The data show several contrasts between noun and verb contexts and targets. First, leaving aside Experiment 4, the verb context data appear relatively consistent across tasks and experiments. Second, there is an asymmetry in syntactic inhibition across contexts: In Experiment 4, where they can be directly compared, nouns are more inhibited in verb contexts than verbs are in noun contexts. These considerations together suggest that, for the kinds of constructions used, verb contexts are syntactically stronger. On the other hand, the noun contexts are semantically stronger (see the large facilitation effects in Experiments 1, 3, and 7). Several factors may contribute here. Referential noun-concepts may be more strongly linked together than they are linked to possible predicates. Verbs may be more constraining in predicate-object constructions (She drove the car) than they are constrained in relative clauses (The car that she...).

Tasks and Representations A widely expressed view is that, in contrast to lexical decision, the naming task is a relatively unbiased measure of lexical access (e.g., Balota & Lorch, 1986). Seidenberg et al. (1984) showed that lexical decision is biased by a number of factors, including syntactic congruence between a single-word prime and a target, by which naming is not influenced. However, using more constraining sentence-frame contexts, West and Stanovich (1986) found that naming latency was affected by the syntactic congruity of contexts and targets. It appears that speakers involuntarily delay the articulation of a word when it is syntactically incongruent with its context. In fact, roughly equivalent effects have been observed in the two tasks in most of the sentence-frame context literature, except that lexical decision tends to show larger overall effects and especially more inhibition of unrelated targets (see Stanovich & West, 1983).

Recently, Lucas, Tanenhaus, and Carlson (1987) have successfully used the lexical decision-naming task contrast to show that instrument inferences are coded in constructed text-level representations. The present data could be interpreted in terms of the same contrast if Experiment 7 were disregarded. That is, it could be claimed that the strong facilitation effects for nouns in noun contexts in the lexical

decision task reflect text-level integration (O'Seaghdha, in press), while the naming data reflect largely lexical influences. However, the statistical weakness of the relatedness effect in Experiment 4 suggested that this conclusion might be premature. Why should the frequently observed robust facilitation of semantically related congruent sentence completions not be replicated here? It turned out that the relatedness effect was nonsignificant in Experiment 4, significant but syntax insensitive in Experiment 5, and of the expected magnitude only in Experiment 7.

Resolution of the status of the naming task may be at hand if a contrast between sentence intrinsic and sentence external targets is taken into account. The lexical decision and naming tasks may validly be used to index different representations when, as in the Lucas et al. study, the probes are not continuations or completions of the sentences being indexed. However, the syntactically sensitive naming task may be a mercurial index when it is used to tap sentence intrinsic effects. This suggestion is supported both by the present results and by previous evidence that relatively minor changes in procedure can dramatically influence naming latency to sentence-completions. For example, slightly different procedures in Stanovich & West's (1979) Experiments 1 and 2 produced a jump from 15 to 111 msec of facilitation.

Conclusions In addition to the foregoing methodological considerations, several substantive conclusions can be drawn from the present data. First, although the nature of these processes requires further study, it appears that relatedness effects in both lexical decision (Experiments 1 to 3) and naming (Experiment 7) on sentence-completion targets can reflect processes of text integration. The mapping between performance on the tasks and on-line reading remains indeterminate. However, the fact that syntactic inhibition was observed in Experiment 4 when relatedness effects were virtually eliminated, suggests that syntactic assignment may be an obligatory process of a kind that syntactic-semantic integration is not.

ACKNOWLEDGMENT

Preparation of this report was supported by NIH Grant NS25502.

REFERENCES

Balota, D. A. & Lorch, R. F. (1986). Depth of automatic spreading activation: Mediated priming effects in naming but not in lexical decision. Journal of Experimental Psychology: Learning, Memory, & Cognition, 12, 336-345.

Fischler, I. & Bloom, P. A. (1979). Automatic and attentional processes in the effects of sentence contexts on word recognition. Journal of Verbal Learning and Verbal Behavior, 18, 1-20.

Forster, K. I. (1979). Levels of processing and the structure of the language processor. In W. E. Cooper & E. Walker (Eds.), Sentence processing: Psycholinguistic studies presented to Merrill Garrett (pp. 27-85). Hillsdale, NJ: Erlbaum.

Lucas, M. M., Tanenhaus, M. K., & Carlson, G. N. (1987). Inferences in sentence comprehension: The role of constructed representations. Program of the Ninth Annual Conference of the Cognitive Science Society, pp 566-574.

O'Seaghdha, P. (1986). Anatomising lexical decision in phrasal contexts. Program of the Eighth Annual Conference of the Cognitive Science Society, pp 773-779.

O'Seaghdha, P. (in press). The dependence of lexical relatedness effects on syntactic connectedness. Journal of Experimental Psychology: Learning, Memory, and Cognition.

Seidenberg, M. S., Waters, G. S., Sanders, M., & Langer, P. (1984). Pre- and postlexical loci of contextual effects on word recognition. Memory & Cognition, 12, 315-328.

Stanovich, K. E. & West, R. F. (1979). Mechanisms of sentence context effects in reading: Tests of a two-process theory. Memory & Cognition, 7, 77-85.

Stanovich, K. E. & West, R. F. (1983). On priming by a sentence context. Journal of Experimental Psychology: General, 112, 1-36.

Tanenhaus, M. K., Dell, G. S., & Carlson, G. (1987). Context effects and lexical processing: A connectionist approach to modularity. In J. L. Garfield (Ed.) Modularity in knowledge representation and natural-language understanding. Cambridge, MA: MIT Press.

West, R. F., & Stanovich, K. E. (1986). Robust effects of syntactic context on naming. Memory & Cognition, 14, 104-112.

Wright, B., & Garrett, M. F. (1984). Lexical decision in sentences: Effects of syntactic structure. Memory & Cognition, 12, 31-45.

Generalization by humans and multi-layer adaptive networks

M. Pavel Mark A. Gluck Van Henkle

Stanford University

ABSTRACT

Generalization of a pattern categorization task was investigated in a simple, deterministic, inductive learning task. Each of eight patterns in a training set was specified in terms of four binary features. After subjects learned to categorize these patterns in a supervised learning paradigm they were asked generalize their knowledge by categorizing novel patterns. We analyzed both the details of the learning process as well as subjects' generalizations to novel patterns. Certain patterns in the training set were consistently found to be more difficult to learn than others. The subsequent generalizations made by subjects indicate that in spite of important individual differences, subjects showed systematic similarities in how they generalized to novel situations. The generalization performance of subjects was compared to those that could possibly be generated by a two-layer adaptive network. A comparison of network and human generalizations indicate that using a minimal network architecture is not a sufficient constraint to guarantee that a network will generalize the way humans do.

INTRODUCTION

Inductive learning is one of the most difficult and least understood aspects of cognition. During supervised learning an organism is exposed to a few examples of stimulus-response pairs (the training set) from which the organism infers how to to generate correct responses to many other stimuli. The theoretical problem arises from the fact that there usually are many rules that are consistent with the training set but which generate different responses to the novel stimuli. Unlike deduction, induction has no *a priori* normative procedure to decide which set of rules is the most appropriate.

Thus, induction problems can be considered ill-posed problems in that there too many very different solutions. Such problems can be solved by introducing additional constraints or objectives that are external to the original problem. One of the central problems for understanding induction in natural (human) or artificial systems is to determine useful constraints or regularization principles that convert the ill-posed problems into well posed problems.

In spite of the inherent difficulties with defining "good" inductions, people appear to be very good at rapidly learning to induce useful rules. Investigation of how people perform induction or generalization is, therefore, interesting not only to the students of cognition but also to builders of artificial learning machines. Although there have been many attempts to study this problem, most of previous research has been focused primarily on investigation and modeling of average performance (Medin & Schaefer, 1978; Medin, Dewey, & Murphy, 1983; Nosofsky, 1986).

Correspondence should be addressed to: M. Pavel, Bldg 420, Stanford, CA 94305. This research was supported by NSF Grant BNS-8618049, NSF Grant IST-8511589 and Grant from NASA Ames NCC-2-307.

One goal of study reported here was to examine how people generalize in a simple deterministic categorization task in which each pattern is characterized in terms of known binary features. While we expected certain similarities to emerge across human learners, we anticipated that the particular generalizations might be subject to considerable individual differences. To test this idea, we used an experimental paradigm that would permit us to to observe individual subjects during the learning of a categorization task on a set of training patterns and then allow us examine the types of categorizations they made on a set of novel test patterns. In the last section of this paper we compare human generalizations to those of a small adaptive network.

EXPERIMENT 1

The purpose of this study was to record subjects' progress in learning a deterministic categorization, analyze their generalizations, and compare their performance to that of small adaptive networks. The stimuli were similar to those used by Medin, Altom, Edelson, & Freko, (1982), but the procedure was designed to enable us to monitor the learning process in addition to evaluating subsequent generalizations.

Method

Seventy-eight Stanford undergraduates were run from a pool of subjects enrolled in an introductory psychology course. Each stimulus item was composed of four binary dimensions and was presented to subjects as a patient chart listing four different symptom types: Muscles (*tense* or *relaxed*), Insulin (*high* or *low*), Glands (*swollen* or *recessed*), and Sinus (*stuffy* or *runny*). The complete stimulus set consisted of the 16 possible patterns resulting from forming all combinations of the four binary dimensions. As shown in Table 1, with the alternate values of each dimension indicated by either "1" or "0", four of the 16 stimuli were designated as members of category A, four as members of category B, and the remaining 8 were presented as novel items to test for generalization.

TABLE 1: Category Structure from Experiment 1

Category	Item	Dimension 1	2	3	4
A	A1	1	1	1	1
	A2	1	1	0	0
	A3	0	1	1	1
	A4	1	0	0	0
B	B1	0	0	1	0
	B2	0	0	0	1
	B3	1	0	1	0
	B4	0	1	0	1
Novel	N1	0	0	0	0
	N2	0	0	1	1
	N3	0	1	0	0
	N4	1	0	1	1
	N5	1	1	1	0
	N6	1	1	0	1
	N7	0	1	1	0
	N8	1	0	0	1

The training stimuli were carefully selected so that the categorization could be performed perfectly by an exclusive-or (XOR) on the last two dimensions (3 & 4) while the first two dimensions (1 & 2) could be used to form a simpler but less effective rule.

Procedure. Throughout the experiment the two categories were referred to as "Turitis" and "Purosis", with the association of each name to a category randomized across subjects. The patient charts were presented on a computer screen with the symptoms arranged vertically (as above). For each individual subject, the order in which the symptom types were displayed on each chart was consistent throughout the entire experiment. However, across subjects the order of display was randomized. The particular symptom names associated with each dimension were also randomized across subjects.

Subjects were instructed to imagine that they were medical interns learning to diagnose patients suffering from one of two diseases. They were told that they would learn to make their diagnoses by attempting to diagnose individual patients: they would be shown a patient chart listing four symptoms, attempt to make a diagnosis, and then be given the correct diagnosis. Subjects were told that they would complete their training after correctly diagnosing approximately 32 patients in a row, at which point they would be tested on their ability to make diagnoses. After being given these instructions, the training phase of the experiment began.

The training phase consisted of successive presentations of the eight stimuli in categories A and B. On each trial one stimulus item was presented, the subject was prompted for a category judgement, and then the subject was given feedback specifying the correct categorization. The order of presentation for the training stimuli was randomized over blocks of 16 trials so that two instances of each item from category A and two instances of each item from category B occurred in each block. The training phase continued until a subject had either met the learning criterion of correctly categorizing all items in two successive blocks, or until the subject had completed 15 blocks without meeting the criterion.

Upon completing the training phase of the experiment the subjects were instructed that they would be tested on the knowledge they had gained in that phase by diagnosing 32 additional patients. They were then presented with 2 instances of each of the 16 stimuli in a random order. For each stimulus they were asked to make their diagnosis and then rate their confidence of the diagnosis on a scale of 1-7 (least to most confident). No feedback was given on these test trials. Following the experiment, subjects were asked to describe the methods they used to make their diagnoses. The entire experimental process took between 30 minutes and one hour, depending on how quickly the subject reached criterion during the training phase.

Results

A summary of the results for all subjects are shown in Figure 1. Each panel of Figure 1 represents the proportion of *A* responses; the first eight patterns represent the training and the last eight the transfer set. The first four patterns of the training set are from category A and the second four patterns from category B. The left panel shows data from the 40 subjects who reached the criterion together with the data of Medin at al (1982). *criterion* subjects learned the task better than those of Medin et al. (1982). Their average performance is almost identical to the results of Medin et al. (1982). In contrast, our *non-criterion* subjects who did not reached the criterion are more similar to Medin's data for the training patterns but differ considerably on the novel patterns. We conclude that our criterion subjects are most like those of Medin except for the more rigorous training given in this experiment.

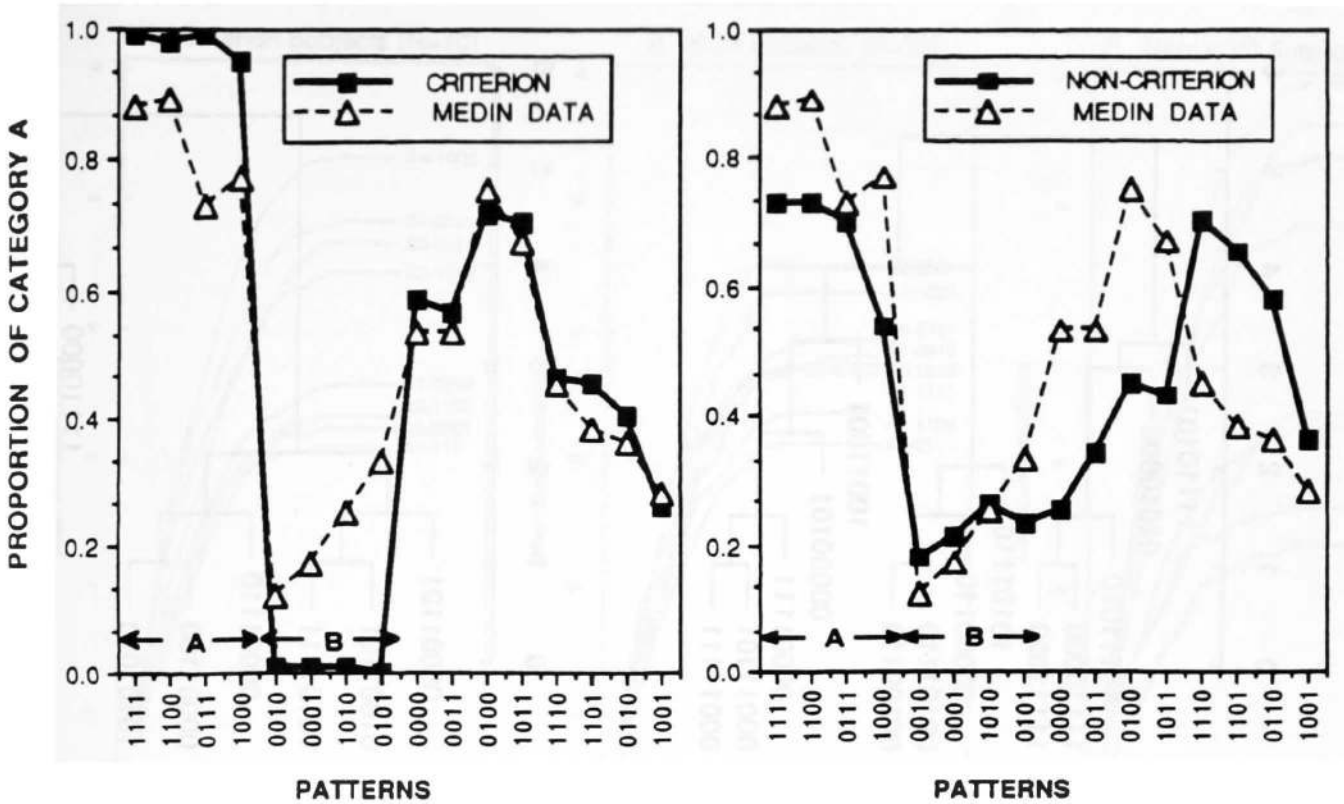
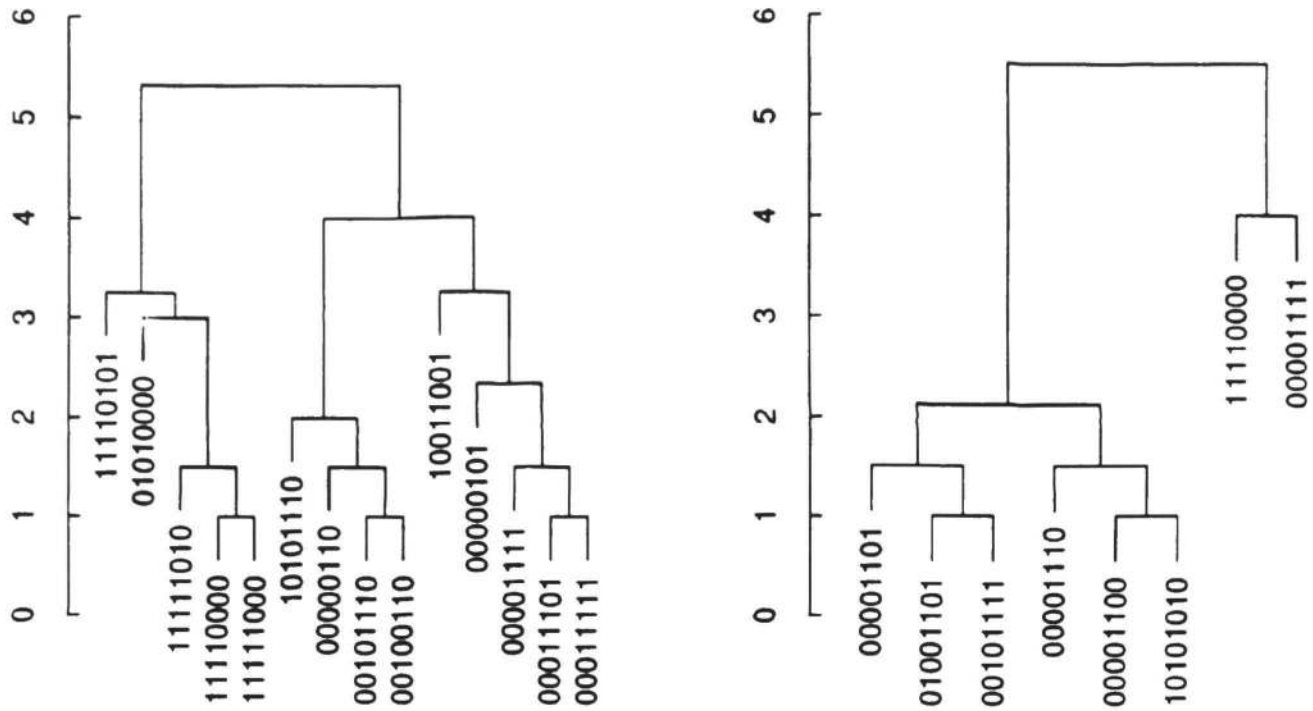


Figure 1. Generalization profiles.

Responses to the novel patterns represent the transfer of learning or generalization performed by subjects. There are several ways of interpreting the proportions of the *A* responses. According to one interpretation, the average responses arise from an ensemble of identically distributed subjects. That is, the probability of assigning a given pattern to category A is the same for each subject and is approximated by the graph on Figure 1. This interpretation is commonly assumed by investigators (e.g. Medin et al., 1982) who used such data to test exemplar-based models of categorization. An alternative way of interpreting these proportions, however, is in terms of a mixture of distributions corresponding to subjects who learned different rules during the training phase. We examined individual differences in order to distinguish these two interpretations.

The extreme version of the mixture hypothesis is that each subject learned a different set of rules. That model is unlikely because although there are 256 different possible generalizations for the eight test patterns, 14 different generalizations accounted for 85% of the subjects. In particular, generalizations of 38% of the subjects who reached criterion were consistent with the hypothesis that subjects based their categorization on the exclusive-or (XOR) of dimensions 3 and 4 (the graph of XOR performance, if plotted on Figure 1, would consist of alternation of four high and four low responses). On the other hand more than a half of the subjects who learned the task generalized differently. This supports the notion that individual subjects abstracted different set of rules during the training phase. The data in Figure 1 appear to represent a mixture of strategies and generalizations.

While subjects produced many different generalizations, it is possible that these generalizations are very similar to each other. The following analysis was performed to determine similarity among



A. Subjects' Generalizations **B. Network Generalizations**

Figure 2. Hierarchical clustering of generalizations

different generalizations. The generalization performed by each subject can be represented as an eight dimensional *generalization profile* vector, where each "1" bit corresponds to assignment of the corresponding pattern to category A. [1] Hence, similar generalizations would have similar profiles. To analyze the generalization profiles we computed Hamming distances between all pairs of the 14 most frequent generalizations and then used hierarchical clustering, based on average inter-cluster distances, to represent the similarities among generalizations. The resulting hierarchy is shown in Figure 2A. The distance between any two profiles, shown as terminal nodes of the tree, corresponds to the lowest common node on the tree. This analysis indicates that different subjects generalized in many, quite different ways.

In order to understand human categorization process it is important to determine how different subjects arrive at different rules. While a complete answer to this question is beyond the scope of this paper one can get some indications of the underlying processes by examining subjects' average performance during the training phase. Subjects' performance on each pattern during the learning phase was summarized by computing the average cumulative error for each training pattern. Three sets of such cumulative error learning curves are shown in Figure 3A for all subjects who reached criterion, for those that performed XOR (Figure 3B) and for the remainder (Figure 3C). The most important aspect of the cumulative error curves is that, more or less consistently over subjects, each pattern is learned with different difficulty. For example, the pattern 1111 from category A was very easy (few errors) while the pattern 1000 from the same category was very difficult. This regularity which was was true for for all

[1] The order of bits corresponds to the ordering of novel stimuli in Table 1,(N1,N2...N8).

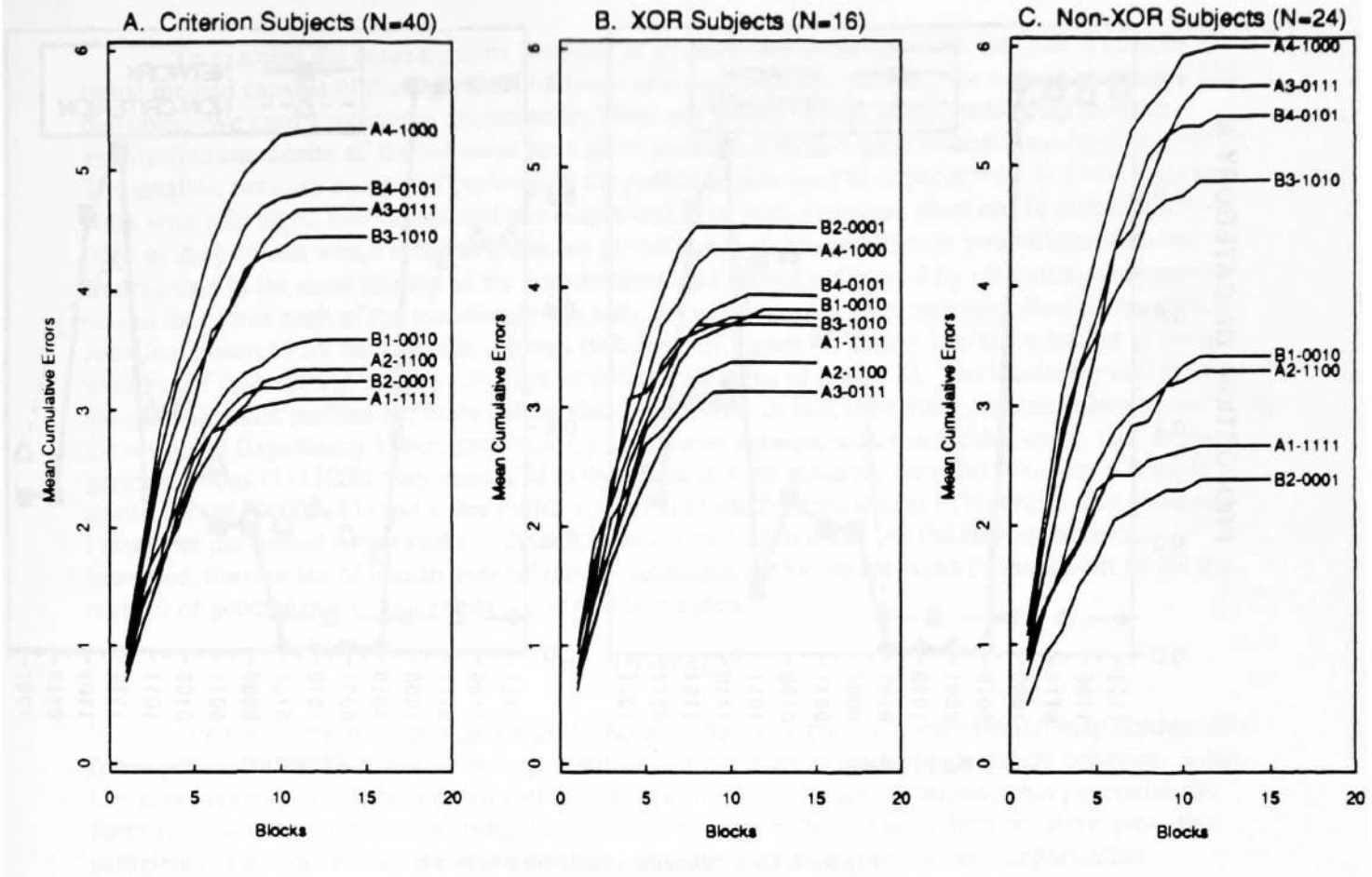


Figure 3. Cumulative error learning curves

subjects who reached the criterion performance is indicative of the type of rules abstracted by subjects. In particular, even those subjects who eventually used the XOR categorization were initially using the first two dimensions.

MODELING GENERALIZATIONS

The empirically observed subjects' generalizations can provide information about the constraints used by human beings. To discover these constraints frequently requires a model-based analysis of the data. Models of categorization can be used in two ways. One approach is based on those models that can represent any generalization and do not impose any prior constraints. Their utility is in remapping the data so that the constraints are easily observed and extracted.

Another way to discover the constraints imposed by the learner is to construct a model of a pattern categorization process that embodies some of those constraints. Such a model can then be used to predict the generalizations and its predictions can be compared to the data.

Generalizations by Networks

An interesting class of models to consider for categorization are multi-layered adaptive networks. Layered networks are acyclic (nonrecursive) directed graphs with defined starting (input) and terminating (output) nodes (units) in which each unit has a uniquely defined distance (in terms of arcs) from all the input units. Hidden units are those nodes that are labeled neither input nor output.

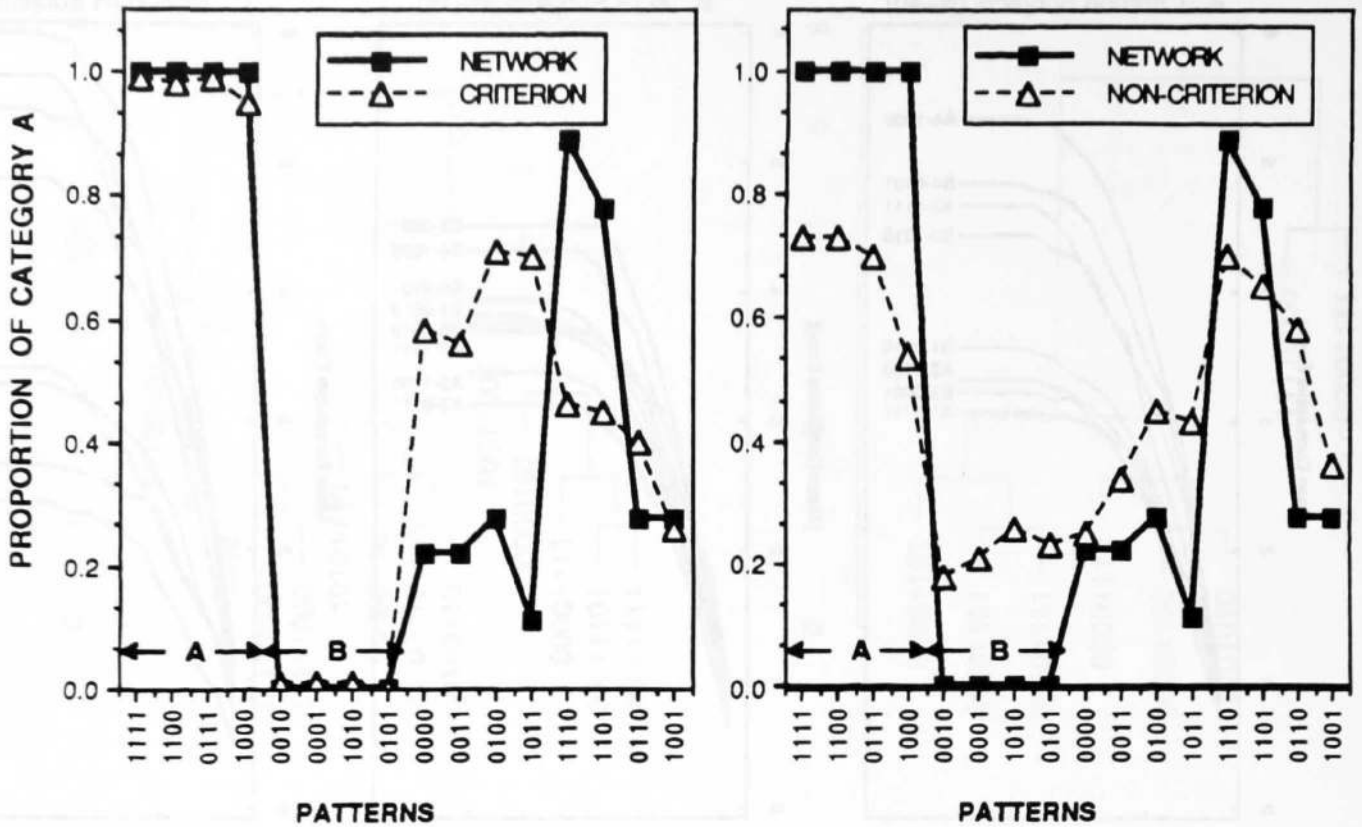


Figure 4. Generalization profile for a network.

Each directed arc is labeled by a real valued weight. A unit may, in general, be a dynamical system but in the current framework a unit is defined by a threshold function of the sum of incoming arcs; the value contributed by each arc is equal to the value of the originating unit multiplied by the weight of the arc. Each unit performs a linear threshold function which is the essential nonlinearity required for a pattern recognition mechanism.

A two-layer adaptive network consisting of an input, hidden and output layer with unlimited number of hidden units can represent any computable boolean function (Nilsson, 1965; Minsky and Papert, 1969). Therefore, such networks can be used to analyze the data by finding a set of weights that performs the same categorization as an individual subject and then examine the structure of such a network.

Because an unconstrained network can make any possible generalization, additional constraints must be imposed if an adaptive network is to predict human generalization performance. An important question to ask is whether or not a network with a specific set of constraints can predict a particular generalization. A complete theory would have to include a characterization of the effects of different constraints on generalization. Although such an analysis is beyond the scope of this paper we illustrate the approach using a particular constraint.

An example of one such constraint involves imposing a limit on the number of hidden units. In the extreme, most constraining case, this amounts to finding an adaptive network with the minimum number of hidden units that can perform the categorization on the training set. The motivation for such an approach is in the usual heuristic arguments for simplicity; a smaller network should generalize better.

To examine the generalization behavior of minimal networks, however, requires a computational method capable of finding all the solutions to a categorization problem for a given number of hidden units. We used a technique developed by Pavel and Moore (1988) using linear programming approach to enumerate all the solutions for a given number of hidden units in small two-layer networks. The smallest network capable of performing the particular task used in Experiment 1 is a two-layer network with four input, two hidden and one output unit. For such a network there are 18 different solutions to the problem which result in 8 distinct generalizations. These different generalizations were summarized in the same manner as the experimental data shown in Figure 4 by computing the proportion of times that each of the test stimuli was assigned to category A. The resulting distribution differs from that observed for the criterion subjects (left panel of Figure 4). In fact, it is more similar to the distribution of responses of the non-criterion subjects (right panel of Figure 4). The clustering analysis of the generalization profiles in Figure 3B are clearly different. In fact, only three generalizations found in the results of Experiment 1 were generated by a two-layer network with two hidden units. One of these generalizations (11110000) corresponded to the frequent XOR solution; the other two were a complement of XOR (00001111) and a rare profile (00101111) which is not shown in Figure 2. The same analyses were performed on networks with larger number of hidden units. As the number of units increased, the number of human generalizations accounted for by the networks increased but so did the number of generalizations not exhibited by human subjects.

SUMMARY

We have demonstrated that subjects who learn the same pattern categorization may abstract different principles and therefore show large individual differences in their generalization behavior. Adaptive networks with the minimum number of hidden units exhibit a similar behavior but generalize differently. Thus, the constraint of using the minimum number of hidden units does not alone provide a sufficient constraint on adaptive network models to allow them to model human categorization processes. Currently we are investigating the effects of other constraints.

REFERENCES

Medin, D. L., Altom, M. W., Edelson, S. M., & Freko, D. (1982). Correlated symptoms and simulated medical classification. *Journal of Experimental Psychology: Learning, Memory, & Cognition, 8*, 37-50.

Medin, D. L., Dewey, G. I., & Murphy, T. D. (1983). Relationships between item and category learning: Evidence that abstraction is not automatic. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 9*, 607-625.

Medin, D. L., & Schaffer, M. M. (1978). Context theory of classification learning. *Psychological Review, 85*, 207-238.

Minsky, M., & Papert, S. (1969). *Perceptrons: An Introduction to Computational Geometry.* Cambridge, MA: MIT Press.

Nilsson, N. J. (1965). *Learning machines.* New York: McGraw-Hill.

Nosofsky, R. M. (1986). Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General, 115*, 39-57.

Pavel, M., & Moore, R. T. (1988). *Computational analysis of solutions of two-layer adaptive networks.* APL Technical Report, Dept. of Psychology, Stanford University.

Pattern-Based Parsing for Word-Sense Disambiguation

Rajesh S. Virkar & John W. Roach
Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061

Abstract

In the study of natural language understanding, the reductionist approach has been commonly used by A. I. researchers. Here we develop a technique for parsing based on this approach. We use a set of semantic primitives to represent word meanings and utilize patterns of sentences for mapping sentences onto meaning structures. To assist the parsing process, we develop semantic mappings for primitive sentences, semantic transformations for decomposing complex sentences using function words and axioms that encode world knowledge. We then explore the application of our approach to the word polysemy problem.

1. Introduction

Automated word-sense disambiguation is an unsolved problem in computational linguistics, and it forms the goal of the work presented here. We develop a pattern-based parsing technique and give illustrations for word-sense disambiguations using only the sentential context. Our solution to this problem utilizes the semantic primitives approach and appears to be a significant step in determining the ultimate consequences of this approach.

We hypothesize that most of the every-day lexicon can be represented by a large, yet finite, number of semantic primitives. Intuitively, we foresee two kinds of connections between these primitives: a 'tangled hierarchy' of primitives and 'semantic connections'. The tangled hierarchy is used to establish properties of an object inherited all the way down from the root-level nodes, while semantic connections are those relationships that capture world knowledge and provide us with clues in understanding linguistic expressions. Sentence forms can now be classified by the patterning of semantic primitives and function words. Function words are certain words (namely, prepositions, conjunctives, etc.) that may appear in a sentence. We hypothesize that all simple sentence forms can be captured by a very large, yet finite, set of patterns.

We employ a multi-tiered approach to parsing linguistic expressions. This approach accounts for the three different representations of expressions, namely 1. the surface representation of sentences, 2. the verbal representation of semantic primitive patterns, and 3. the deep meaning representation. Processing, in this approach, involves transforming the surface structure into its deep meaning representation.

2. Semantic Primitives and Their Relationship to Meaning

The set of semantic primitives we use is quite large (over 120 semantic primitives). These primitives can be broadly grouped under four classes: ENTITY, EVENT, ABSTRACT and RELATIONAL, and this categorization forms the foundation of the deep meaning representation, as will be seen later in the 'sentence meaning structures'. This classification scheme is adapted from Nida's work on componential analysis (Nida, 1975).

The primitives approach to developing a lexicon has been used by several computational linguists, including Laffal (Laffal, 1973). Laffal's work resulted in a set of 118 semantic categories that were combined to express around 23,500 words of the English language. Each word in this 'concept dictionary' was expressed by one category or by concatenating at most two categories.

Definition 1: A semantic primitive is a set that represents a class of words that refer to the same concept.

Example semantic primitives of the class ENTITY are PERSON (e.g. John) and WRITING-ARTIFACT (e.g. pen); examples of EVENT are DIRECTIONAL-MOTION (e.g. come) and DISTRIBUTE (e.g. give); examples of ABSTRACT are HEALTH-N (e.g. polio) and INTELLECT (e.g. smart); while examples of RELATIONAL are SPATIAL (e.g. inside) and TEMPORAL (e.g. before).

Definition 2: Interpretation 'I' refers to mapping a lexeme onto one of the possible senses of that lexeme. Here, we are acknowledging that a word or a sentence can have several interpretations. Senses are represented by semantic primitives and hence, we can say that $I : w \rightarrow P$. The set of interpretations $\mathbf{I}_w$ of a word **w** is equal to the set of semantic primitives $\mathbf{P}_w$ that can represent the given word in different situations.

Definition 3: The meaning of a word is the intended sense of the word in a given sentence. Thus, the meaning of a word **w** in a given sentence refers to one interpretation i_x represented by the semantic primitive $\mathbf{P}_x$ in the set $\mathbf{P}_w$ (i.e. $\mathbf{P}_x \varepsilon \mathbf{P}_w$). Similarly, the meaning of a sentence is the intended sense of the sentence.

Primitive Cohesion refers to that aspect of language which enables us to create newer (and/or larger) meaningful expressions by combining words. For example, by knowing the meanings of words such as 'John', 'drinks' and 'water', we can combine them to create a meaningful expression such as 'John drinks water'. Meaning has the property of preserving cohesion, i.e. meaning is that sense of a word that conforms to the semantic and linguistic constraints and contributes to the meaningfulness of the sentence.

Definition 4: A meaningful expression is an expression that describes an event with the help of other basic classes such as entities, abstracts and

relationals. It is implied that to describe an event, one or more of these basic classes may not be required. A meaningful expression is neither anomalous, indeterminable, nor contradictory (Allan, 1986).

Suppose that we denote the set of all semantic primitives by Π. Then, Π^+ denotes the set of all strings, of length ≥ 1, of semantic primitives. The language L, the set of all meaningful expressions, can now be viewed as a subset of the set of all possible strings derived from Π. In other words, $L \subset \Pi^+$. Natural language understanding can now be roughly defined as a mapping, U, that translates all meaningful expressions in L onto the set of "sentence meaning structures" (SMSs), S; i.e., $U: L \rightarrow S$.

Definition 5: A function word is a word that, in its position in the sentence, signals the beginning of a new (possibly primitive) meaningful expression. For example, 'Jack and Jill went up the hill'. The set of function words **F** contains logical connectives such as 'and' and 'or', and prepositions such as 'for', 'of', etc.

Definition 6: A primitive sentence is a sentence that conveys only one "meaningful expression" (definition 4) in a language. A complex sentence, on the other hand, is a sentence that expresses two or more (related) meaningful expressions connected by one or more function words.

A simple complex sentence is one that has two meaningful expressions connected by one function word. Hence, by recognizing the function word, the two expressions can be separated into two primitive sentences.

Development of some sort of a mapping scheme is essential to achieve the deep meaning representation from the verbal representation. This mapping scheme should accept an incoming sentence and its pattern, and return an SMS (sentence meaning structure) for that sentence.

Definition 7: A semantic mapping is a function that translates a sentence, based on its corresponding semantic pattern, into one of the possible interpretations. An example of a mapping can be given as follows:

σ_{s1} : PERSON DIRECTIONAL-MOTION DIRECTION-RELATIONAL ARTICLE LOCATION

→	[SMS	**(ENTITY**	PERSON)	
		(EVENT	*DIRECTIONAL-MOTION*)	
		(RELATIONAL	LOCATION)	]

σ_{s1} can be applied to a fairly large set of primitive sentences but we contend that the same meaning is expressed by all sentences belonging to this set. Here, the three slots (in bold) correspond to 3 of the ~120 primitives. It can be shown that any meaningful primitive natural language sentence can be reduced to its meaning in the form of an SMS.

Definition 8: An axiom is a rule based on the pattern of semantic primitives representing a sentence that allows the correct translation of the sentence into its corresponding meaning structure. An axiom helps the parsing process in eliminating the incorrect interpretations of a sentence. Execution of a semantic mapping must be guided by axioms to realize the

correct meaning representation.

Let $S= w_1 w_2 \ldots\ldots w_r$ be a sentence with r words and $P= p_1 p_2 \ldots\ldots p_r$ be the pattern of semantic primitives based on the sentence S. An axiom a_i, as it governs the execution of a semantic mapping m_j, can be given in terms of S and P as follows:

$$a_i[m_j(w_1 w_2 \ldots\ldots w_r, p_1 p_2 \ldots\ldots p_r)] \rightarrow \alpha(w_k, p_k, <<w_1 w_2 \ldots\ldots w_r>>), \forall k \in \{1,2, \ldots\ldots ,r\}$$

Here, α is an admissibility procedure that checks the validity of the selected interpretation of a word in the context of the presented sentence. The enclosing symbols (<<,>>) are used to represent the sentence as a single entity, i.e. a meaningful expression consisting of a string of words.

$$\alpha(w_i, p_i, <<w_1 w_2 \ldots\ldots w_r>>) \rightarrow <<w_1 w_2 \ldots\ldots w_r>>_{wi,pi} \parallel p_i \subset \chi, \text{ where } \chi \in \Pi$$

Here, '$\parallel$' is a formal relation named 'is supported by', and the formula is read as follows: for the ith word (w_i) in the sentence $<<w_1 w_2 \ldots\ldots w_r>>$, the instantiation of the chosen semantic primitive (p_i) that represents its meaning is supported by the set inclusion property that relates p_i to one of the four basic classes of semantic primitives. The sentential context is used for guiding α in a given axiom.

The set of all axioms **A** can be developed by adding a new axiom (or rule) for every new interpretation of a word, produced in different patterns of primitives containing at least one primitive that represents that word. In other words, each axiom in **A** represents "primitive cohesion" or the links between primitive concepts.

We can claim that if $\mathbf{i_x}$ is a newly encountered (from the standpoint of **A**) interpretation of a word $\mathbf{w_k}$ in a sentence **S** (represented by a pattern **P**), and that an axiom governing such a semantic mapping does not exist, then the set **A** can be augmented by adding the appropriate axiom that oversees this semantic mapping.

<u>Definition 9:</u> A <u>semantic transformation</u> can be defined as the process of decomposing a complex sentence into two primitive sentences using the function word appearing in the complex sentence, without altering the 'meaning' expressed by the original complex sentence.

Let **T** be the set of semantic transformations and t_i be some semantic transformation. Let w_j ($j \leq m$, m is the length of the sentence S) be the function word. This means that S is a simple complex sentence that is a combination of two meaningful expressions. Then we can say that

$$\exists\, t_i \in T \text{ such that } t_i(S,P,w_j) \Rightarrow < (S_1,P_1)\ (S_2,P_2) >$$

where $< (S_1,P_1)\ (S_2,P_2) >$ preserves the meaning expressed by (S,P).

It should be noted, however, that several semantic transformations may exist for a given function word, and the one that is applied is selected based on the semantic pattern of the sentence. It should also be noted that a semantic transformation on a sentence that does not

contain any function words is equivalent to applying a semantic mapping.

It is possible that after applying a semantic transformation, one or both of the resulting sentences will be complex. This can occur only in the presence of more function words. These function words can now be used to apply other semantic transformations and further reduce the sentences. Since every execution of a semantic transformation reduces the complexity of the sentence, a finite number of semantic transformations guarantees convergence to primitive sentences.

3. Word-Sense Disambiguation

Word-sense disambiguation has been a very important aspect of natural language understanding. From the definition of our semantic primitives, we know that different meanings of any given word refer to different corresponding semantic primitives as interpretations. Thus, polysemy is now reduced to the problem of removing extraneous/illegal interpretations from the set of all possible interpretations until we have reached the appropriate meaning.

A typical case of polysemy occurs in a class of sentences where some word may be used in several different places in a sentence. Here, different interpretations of the word suit different positions in the sentence and, in those positions, help represent a coherent meaning structure. This idea of **meaning distribution of multiple senses** will be explained through the following example.

(1) The pen is in the box.
(2) The box is in the pen.
(3) The ink is in the pen.
(4) The bull is in the pen.
(5) The pen is in the pen.

The analysis of these sentences gives a common meaning structure based on relationships of the constituent words and their interpretations. Since the ambiguous interpretations have not been resolved yet, the structure (not fully instantiated) can be given as:

<u>Common Meaning Representation</u>:

```
[SMS        (STATE      is)
            (?a1        ?a2)
            (RELATION   in)
            (?b1        ?b2)                    ]
```

where "?" suggests an uninstantiated variable.

It is obvious that 'pen' is the ambiguous word in the group of sentences (1) through (5). The corresponding common interpretations are a. a pen as a writing instrument, and b. a pen as a small construction used for confinement. Axioms can now be applied to resolve the ambiguity in the meaning of 'pen'. The incorporation of semantic constraints and common world knowledge into these axioms should then be evident. Note that the following analyses are constructed using only the sentential context.

Analysis of (1):

$a(S_{p1}.S_{p2}... S_{p6}$, [The pen is in the box])

resolves 'pen'

using the function $\alpha(S_{p2},[pen])$

α in turn validates a possible instantiation-

Check_Instantiation(WRITING-ARTIFACT, [pen])

The reasoning can be presented as follows: a confining construction cannot be contained in a box, while a writing instrument can be contained in a box. This results into the resolved meaning structure:

```
        [SMS    (STATE              is)
------>         (WRITING-ARTIFACT   pen)
                (RELATION           in)
                (CONTAINER          box)        ]
```

Similar analyses can be performed on sentences (2) through (4). The last sentence, (5), is a very interesting example (due to Bar-Hillel). It involves two occurrences of the word 'pen' in which the object referenced by the meaning of one occurrence is contained inside the object referenced by the meaning of the second occurrence. Now, let us look at its analysis.

Analysis of (5):

$a(S_{p1}.S_{p2}... S_{p6}$, [The pen is in the pen])

resolves the occurrences of 'pen'

using the functions $\alpha(S_{p2},[pen])$ and $\alpha(S_{p6},[pen])$

α in turn validates two possible instantiations-

Check_Instantiation(WRITING-ARTIFACT, [pen]) ----> for S_{p2}

and **Check_Instantiation**(CONSTRUCTION, [pen]) ----> for S_{p6}

The reasoning here involves the world knowledge about the sizes of the two 'pens'. An instance of a construction cannot be contained inside a writing instrument, while a writing instrument can certainly be kept in a storage construction. We are assuming that any object cannot be stored within itself, and in the event that a smaller 'pen' is kept inside a larger 'pen' (of the same interpretation), the context will dictate the choice. This reasoning results into the resolved meaning structure:

```
        [SMS    (STATE              is)
------>         (WRITING-ARTIFACT   pen)
                (RELATION           in)
------>         (LOCATION           pen)        ]
```

From the example sentences it should be clear how axioms incorporating semantic constraints can be used to disambiguate multiple senses involving meaning distribution in the same sentence.

4. Results

A natural language parser developed using our approach of semantic primitives and their patterns is operational on a VAX 11/785 with a 1 KLIPS Prolog interpreter. The examples presented in this paper are actual results among many produced by the program.

The lexicon used by the program, in its current state, has over 500 different word senses. The words are divided into four broad classes: (a) Entities, (b) Events, (c) Abstracts, and (d) Relationals. There are over twenty different semantic transformations in the program based on function words and phrases. The transformations support over forty

semantic mappings. The program also contains over forty axioms that aid in reducing sentences into their respective sentence meaning structures.

The techniques described in this paper have been used to develop a natural language knowledge assimilation system for expert systems (Virkar & Roach, 1988). This system augments an already existing knowledge base of a drug interaction expert system, DIE (Roach, et. al, 1985), by reading and understanding not only sentences but entire research paper abstracts in the pharmacology domain thus demonstrating the extension of these techniques to the text-level.

5. Conclusion

In this paper we have developed a pattern-based technique for natural language parsing based on semantic primitives and their patterns. We have developed axioms from the semantic primitives for encoding world knowledge.

We have given examples of word-sense disambiguation in sentences using only the sentential context under the class of meaning distribution. We have shown, in detail, how the technique is used to arrive at the correct meaning representations.

Future efforts will focus on encoding entailment rules that represent deep inferencing techniques and semantic context structures that dynamically evaluate the parsing process.

Bibliography

Allan, K. 1986. Linguistic Meaning: Vol. 1. New York: Routledge, Kegan Paul.

Goguen, J. A., Thatcher, J. W. and Wagner, E. G. 1978. *An Initial Algebra Approach to the Specification, Correctness and Implementation of Abstract Data Type*. In Yeh, R. T. (ed.) Current Trends in Programming Methodology, Vol. 4. Englewood Cliffs, NJ: Prentice-Hall, Inc.

Guttag, J. V. 1975. *The Specification and Application to Programming of Abstract Data Types*, Ph. D. Thesis, University of Toronto, Department of Computer Science, available as report CSRG-59.

Laffal, J. 1973. A Concept Dictionary of English. New York: Gallery Press.

Nida, E. A. 1975. Componential Analysis of Meaning: An Introduction to Semantic Structures. The Hague: Moulton.

Roach, J., Lee, S., Wilcke, J. and Ehrich, M. 1985. *An expert system for information on pharmacology and drug interactions*. Computers in Biology and Medicine. Vol. 15, No. 1.

Virkar, R. and Roach, J. 1988. *Direct assimilation of expert-level knowledge by automatically parsing research paper abstractss*International Journal of Expert Systems (in press).

MULTIPLE THEORIES IN SCIENTIFIC DISCOVERY[1]

Donald Rose
Department of Information & Computer Science
University of California, Irvine CA 92717 USA

INTRODUCTION

In this paper we describe enhancements being made to REVOLVER, a program that formulates componential models of objects (e.g., physical substances) and replicates historical discoveries from domains such as chemistry and physics. The program inputs reactions relating groups of substances and uses heuristics to transform these premises into models. If premises lead to inconsistent beliefs, the system searches the space of revised premises in order to resolve the errors. The system was originally designed to process only one theory at a time (i.e., to keep only one theory in its database), using hill climbing as its search strategy; once a belief was revised in response to an error, the old belief was deleted.

The enhancements described in this paper deal mainly with extending the REVOLVER framework to handle multiple theories. First, the enhanced system explicitly represents assertions as well as an agent's belief in assertions (meta-assertions). Second, inferencing is performed only on meta-assertions, allowing separate theories to form for multiple agents. By explicitly separating belief in an assertion from the assertion itself, independent theories can coexist for several agents, even though the inference rules used by each can be the same. Third, meta-assertions can indicate how strongly an assertion is believed; this degree of belief could then be used to perform inferencing on strongly-held beliefs first, while it can also be used to bias the system such that they are revised last. The other main enhancement involves explicit representation of unknown objects in system beliefs, whereas the original program only dealt with objects already known to be part of its beliefs.

In the following pages we describe REVOLVER's basic inference and revision processes, then discuss the proposed enhancements to the program. An example from the history of science is then presented which illustrates how the new enhancements can be utilized. Finally, related work plus ideas for future improvements are discussed.

THE REVOLVER SYSTEM

Reactions and *models* are the two kinds of beliefs used by the program. Reactions represent relations between objects and are given as input premises. A premise might represent the inputs and outputs of a chemical reaction. An example from 18th century chemistry would be the observation that potassium and oxygen react to form caustic-potash and water, or (1) K O $\rightarrow$ P W. Another observation from that era was that potassium reacts to form caustic-potash and hydrogen, or (2) K W $\rightarrow$ P H W. Given premises such as these, REVOLVER tries to infer new models of substances by using a set of general heuristics. Since W appears on both sides of (2), it is reduced from the reaction, leaving (3) K $\rightarrow$ P H. Whenever a substance is alone on one side of a reaction, the system infers that its components are present on the opposite side. Thus REVOLVER now infers from (3) the model (4) K = P H. The program then substitutes K's components from (4) into (1) to get

[1]This research was supported by Contract N00014-84-K-0345 from the Information Sciences Division, Office of Naval Research. I would like to thank Pat Langley, Paul Thagard, Randy Jones and Bernd Nordhausen for discussions that helped develop and refine the ideas in this paper.

(5) P H O → P W. Next, P is reduced from both sides of (5), leaving (6) H O → W. The final inference is the model (7) W = H O.

In the above example, the premises were consistent; REVOLVER reached a quiescent state without inferring any erroneous beliefs. However, sometimes the premises given to REVOLVER lead to reactions having either no inputs or no outputs. In order to restore premises to consistency, REVOLVER invokes its belief revision process. The program finds the premises that led to the inconsistency, and considers revisions to them that will lead REVOLVER closer to a consistent set of beliefs. After revising a premise, the system continues making new inferences and, if it detects new inconsistencies, again revises premises. This cycle continues until no more inferences can be made and no inconsistencies exist.

Continuing our example, suppose the system receives new premise (8) P → K O. Substituting for K leads to (9) P → P H O and reducing P yields inconsistent reaction (10) nil → H O. Invoking belief revision, REVOLVER identifies premises (2) and (8) as the sources of the error, and must decide how to revise them in order to resolve the inconsistency.

REVOLVER now tries to eliminate each substance in the inconsistency, one at a time. Hence, in our example, the system proposes six candidate revisions:

R1: add H to (2)'s inputs; R2: add O to (2)'s inputs; R3: delete H from (2)'s outputs;
R4: add H to (8)'s inputs; R5: add O to (8)'s inputs; R6: delete O from (8)'s outputs.

Implementing any of these revisions would remove *one* substance from the inconsistency after REVOLVER has made the change selected, deleted impacted beliefs, and restarted its basic inference process.

Only one of the candidate revisions in REVOLVER is actually carried out. Thus, selecting the best revision(s) is especially important; the program uses an *evaluation function* to make the selection. REVOLVER scores the premises considered for revision along several criteria, multiplies each score by a weight (indicating the priority given to each criterion), sums the weighted scores, and revises the premise(s) having the lowest total score. Since the program does not retain alternate revised premises after each revision step, it is a hill-climbing system, relying on its evaluation function to intelligently guide search towards consistent beliefs. The criteria for selecting the best revision are discussed elsewhere (Rose, 1988; Rose & Langley, 1988).

Continuing our example, let us assume that R5 is selected. This results in revised premise (11) P O → K O. Substituting P and H for K in (11) and reducing P and O yields (14) nil → H. Now REVOLVER proposes revisions again:

R7: add H to (2)'s inputs; R8: delete H from (2)'s outputs; R9: add H to (11)'s inputs.

Now let us suppose R9 is selected. This results in (15) P H O → K O. After making this revision and restarting the basic inference process, substitution leads to P H O → P H O, which after reductions leads to nil → nil, indicating consistency has been reached.

ENHANCING REVOLVER

While the example just seen is a reasonable model of how 18th century chemists reasoned about reactions, there were a number of drawbacks in the REVOLVER framework as presented above. The first main drawback was that it did not reason about multiple theories. Constructing inferences from premises asserted (or recognized) in one order can sometimes lead to a different theory than when the same premises are asserted in some different order. However, the program could only

investigate one of these possibilities. In addition, multiple theories could not coexist in its database. The second main drawback of the system is that it did not separate belief in assertions from the assertions themselves. The third main drawback of REVOLVER is that it did not reason about degrees of belief. That is, one assertion was never believed more than any another.

This paper discusses techniques being developed to enhance the system so that each of these areas are addressed. First, the enhanced system explicitly represents assertions as well as an agent's belief in assertions (meta-assertions). Second, inferencing is performed only on meta-assertions, allowing separate theories to form for multiple agents. Third, meta-assertions indicate how strongly an assertion is believed; this degree of belief can then be used to perform inferencing on strongly-held beliefs first, while it can also be used to bias the system such that they are revised last.

Inferring Multiple Theories

The new version of REVOLVER not only represents assertions, but also represents agents' belief in those assertions. I call the latter *meta-assertions*. For example, an assertion might be the observation (2) K W → P H W. The meta-assertion that agent1 believes (2) would be agent1(K W → P H W). The separation of belief in assertions from the assertions themselves enables multiple theories to coexist in REVOLVER's database. Theories can be created for each agent, each theory being independent of the other.[2] These mutually independent theories result because inferencing in the enhanced REVOLVER is done on meta-assertions only. In other words, beliefs must be recognized by an agent before a new inference can be made; the new inference is then automatically assumed to be held by that same agent.

This enhancement means that even if two agents use the same premises and the same inference rules, different theories may result. This is due to the effects of assertion ordering on the inference process. The order in which agents recognize premises influences the order in which inferences are made, and since different orderings can lead to different theories, it is possible for agents to hold different theories even if their initial premises are the same. In the original REVOLVER, order was never an issue; the most recently asserted premises were always processed immediately, and any new premises were processed as they arrived. The new ability to explicitly recognize premises in any order gives agents the ability to reason about existing premises in different ways, possibly leading to different theories which could then be judged along various dimensions.

Another effect of allowing multiple theories to coexist in REVOLVER is that the same premise might be revised in different ways by different users. Each user can use a unique evaluation function in the new system, embodying different revision strategies and preferences; this means that two users might not revise a premise in the same way. Hence, more than one revision might now be added to the database, although only one of these revisions would be believed by each user. While the same general hill-climbing strategy is still used for each user, each may take unique paths, leading to multiple theories.

Integrating Degrees of Belief and the Evaluation Function

Each user determines the order in which the program recognizes premises. There are two ways of accomplishing this. The user can simply recognize each premise incrementally, waiting for processing to stop before a new premise is recognized. The other way is to explicitly set the degree of belief of each premise; the system would then process premises with highest belief first. Concerning the revision process, degrees of belief can be integrated into REVOLVER's existing

[2] A later section discusses a plausible exception to this.

evaluation function as well. Just as other biases are part of this function, a bias concerning which assertions are held more strongly than others can also be used. That is, while the other measures try to implicitly decide which premise is best suited for revision, the degree of belief measure can be used to explicitly bias the system for or against the revision of certain premises.

Postulating New Objects and Models

Another new enhancement to the system involves the ability to postulate the existence of new objects (e.g., substances) in premises during the revision process. In fact, the use of unknowns can form the basis for a more general approach to revision generation: whenever additions to premises are proposed during revision, it should be possible to simply use an unknown symbol (e.g., X) and then try to resolve the identity of this unknown after further inferencing.

Resolving an unknown can sometimes be done by comparing it to other beliefs in the database. For example, if an unknown X has components C..., while a known object M also has components C..., the system can plausibly infer that X == (is equivalent to) M. Another less certain method of resolving unknowns can come by declaring two unknowns as equivalent when such an act would lead to a new model being inferred. For example, if unknown X has components C..., while unknown Y = M, the system could infer that Y == X, leading to the new model M = C....

I mentioned that theories are generally mutually independent. However, it can be useful to use beliefs inferred during the construction of one theory to influence the construction of a new theory. For example, if the components of an unknown object X in theory1 match the components of a known object M in theory2 (i.e., X = C... and M = C...), then a plausible inference would be to equate X with M, even though M = C... was never inferred during the making of theory1.

Finally, note that when the same inferences are made via different methods, the plausibility of such inferences should increase (i.e., it should become less risky to make such inferences). For example, the inference that M = C... is such a case (from the scenario just presented above).

EXAMPLE

Let us now look at a more complete example of the new ideas outlined above. We saw earlier how the original REVOLVER would handle a case from the history of chemistry involving three initial premises. However, there are order effects in this example; processing premises in a different order can lead to another theory being constructed. Thus, this example (which models the 18th century dispute between chemists Gay-Lussac and Thenard and their contemporary Davy) serves as a good example of the new concepts being presented in this paper. First we will see how two different theories can arise from the same premises, illustrating new concepts along the way. Second, we will see how these theories compare along different dimensions.

Gay-Lussac and Thenard vs. Davy

The example presented at the start of this paper essentially captures the theory held by Gay-Lussac and Thenard. The inference process in the enhanced version of REVOLVER can be represented as follows:

Meta-Assertions: Number and Explanation:
GLandT(K W → P H W) 1 (premise)
GLandT(K → P H) 2 (reduction)
GLandT(P W → K O) 3 (premise)

GLandT(K = P H) 4 from 2 (infer-components)
GLandT(P W → P H O) 5 from 4 and 3 (substitution)
GLandT(W → H O) 6 from 5 (reduction)
GLandT(W = H O) 7 from 6 (infer-components)
GLandT(P → K O) 8 (premise)
GLandT(P → P H O) 9 from 8 and 4 (substitution)
GLandT(nil → H O) 10 from 9 (reduction)

At this point, the system must perform revision. One of the revisions generated explains the inconsistent reaction by postulating an unknown substance in the inputs of 8 (i.e., P X → K O). Suppose this revision is chosen as best. Inferencing would then proceed as follows:

GLandT(P X → K O) 11 (revision)
GLandT(P X → P H O) 12 from 11 and 4 (substitution)
GLandT(X → H O) 13 from 12 (reduction)
GLandT(X = H O) 14 from 13 (infer-components)
GLandT(X == W) 15 from 14 and 7 (equate-models-with-same-components)

This last inference is made when the system tries to resolve X, and finds that water has the same components as X. Hence, X is inferred to be equivalent to W. In other words, the unknown substance in the inputs of Davy's observation P → K O is deemed to be water in the theory of Gay-Lussac and Thenard. This agrees with what took place historically (Zytkow & Simon, 1986).

However, the new system can also model the reasoning Davy used in forming his counterargument to the theory proposed by his two colleagues. The difference in reasoning begins with which premises Davy would hold with highest belief, starting with his observation P → K O. He would then process the other two premises one at a time, trying to fit each into his evolving theory:

Meta-Assertions: Number and Explanation:
Davy(P → K O) 1 (premise)
Davy(P = K O) 2 (infer-components)
Davy(K W → P H W) 3 (premise)
Davy(K → P H) 4 (reduction)
Davy(K → K O H) 5 from 4 and 2 (substitution)
Davy(nil → O H) 6 from 5 (reduction)

Now revision must be performed; the scenario is similar to the case seen earlier for Gay-Lussac and Thenard. Again, one of the revisions proposes an unknown (call it Y) in a premise's inputs to explain the inconsistent reaction. This revision is K W Y → P H W. Suppose this is selected as the best revision to make. Inferencing now proceeds:

Davy(K W Y → P H W) 7 (revision)
Davy(K Y → P H) 8 (reduction)
Davy(K Y → K O H) 9 from 8 and 2 (substitution)
Davy(Y → O H) 10 from 9 (reduction)
Davy(Y = O H) 11 from 10 (infer-components)

At this point, the next inferencing step depends on how much interaction between theories is allowed to occur (a characteristic that should depend on the agent doing the inferencing). If an agent is aware of inferences made within other theories, this can sometimes facilitate inferences within *his* theory that would not otherwise be possible. For example, in our current situation, the

system could realize that a model for water has been inferred within another theory (i.e., W = H O within the theory representing Gay-Lussac and Thenard), and notice that W's components match those of the unknown Y. Hence, Y could be equated to W, thus resolving Y.

Let us suppose that such interaction is not used, and see how inferencing proceeds. The third premise is now finally processed:

Davy(P W → K O) 12 (premise)
Davy(K O W → K O) 13 from 12 and 2 (substitution)
Davy(O W → O) 14 from 13 (reduction)
Davy(W → nil) 15 from 14 (reduction)

At this point, the system must perform revision again. One of the revisions generated explains the inconsistent reaction by postulating yet another unknown substance (call it Z), this time in the outputs of 12: P W → K O Z. Let us suppose that this revision is chosen as best; inferencing would then proceed as follows:

Davy(P W → K O Z) 16 (revision)
Davy(K O W → K O Z) 17 from 16 and 2 (substitution)
Davy(O W → O Z) 18 from 17 (reduction)
Davy(W → Z) 19 from 18 (reduction)
Davy(Z = W) 20 from 19 (infer-components)
Davy(Z == Y) 21 (assume-equivalent-unknowns)
Davy(W = H O) 22 from 21, 20 and 11 (equate-unknowns'-components)

These last two inferences show the other way in which unknowns can be resolved: by trying to equate unknowns to each other. Such an assumption is useful if new models can be inferred as a result, and this is indeed the case here. That is, when the unknowns Y and Z are equated in this example, the system can then equate the unknowns' respective components; since (11) Y = O H, and (20) Z = W, the system infers from the assumption (21) Z == Y that (22) W = H O. Note that this is the same model for water that would have resulted earlier, if the system had equated Y with the W in Gay-Lussac and Thenard's model for water. The fact that two forms of inferencing would lead to the same result makes the belief in W = H O even more plausible here (in the theory representing Davy).

Note that three different types of beliefs are all existing simultaneously in the database: the assertions, the meta-assertions representing beliefs of Gay-Lussac and Thenard, and the meta-assertions representing beliefs of Davy.

Comparing Theories

Once multiple theories can coexist in the system's database, it becomes desirable to develop criteria for comparing such theories. In the example just presented, we can make some intuitive judgments. The theory representing Gay-Lussac and Thenard seems more plausible on at least three dimensions. First, only one revision was needed to reach consistency, while Davy's theory needed two. Second, the model for water was directly inferred in the former theory, while it had to be inferred indirectly in the latter theory. Third, the former theory simply required less inferencing steps overall. Also note that these disparities between the two theories become even greater when other beliefs are added (e.g., further revisions are needed when other premises are integrated into Davy's theory, whereas they can be integrated without revision into his colleagues' theory).[3]

[3]Adding two more 18th century observations (potassium and ammonia reacting to form hydrogen and a substance

Indeed, the view of Gay-Lussac and Thenard did win out historically, and it was found that Davy's observation did overlook the presence of water in his input substances.

DISCUSSION

We have seen that while the new version of REVOLVER still hill climbs for each individual agent (i.e., only one theory is ever kept for any agent), several theories can now coexist in the system's database. Assumption-based truth maintenance systems, or ATMS (de Kleer, 1984), embody a similar approach. However, neither the ATMS nor similar systems (Doyle, 1979) address the issues of generating and selecting plausible revisions (e.g., in reaction-oriented domains such as chemistry and physics). REVOLVER was originally designed to handle both of these tasks, as well as the problem solving tasks involved in scientific discovery. Another approach to multiple theories in scientific reasoning is the ECHO system (Thagard, 1988), which models how multiple theories can develop from evidence, and how the coherence of each theory can be measured in order to evaluate and compare such theories. However, as is the case with TMS systems, ECHO does not address how or when revisions to evidence should be made. Still, Thagard's techniques for judging the explanatory coherence of theories may serve as a valuable guide for future improvements in the REVOLVER framework.

Concerning the future state of the system, the integration of degrees of belief still remains to be implemented. In addition, automatic belief in newly inferred assertions was assumed in this paper; one could imagine a more cautious mode whereby new inferences could be added to the database, but an agent would have to explicitly acknowledge that he/she wished to believe them. This cautiousness could also apply to newly created revisions; in fact, in this latter case such caution might be even more appropriate. In summary, by providing a framework for discovery, revision, and now multiple theory creation, the REVOLVER system seems to provide a solid foundation for the continued exploration of how science evolves.

REFERENCES

de Kleer, J. (1984). Choices without backtracking. *Proceedings of the Fourth National Conference on Artificial Intelligence* (pp. 79–85). Austin, TX: Morgan Kaufmann.

Doyle, J. (1979). A truth maintenance system. *Artificial Intelligence 12*, 231–272.

Rose, D. (1988). Discovery and belief revision via incremental hill climbing. *Proceedings of the International Workshop on Machine Learning, Meta-Reasoning and Logics* (pp. 129–145). Sesimbra, Portugal.

Rose, D. & Langley, P. (in press). A hill-climbing approach to machine discovery. In *Proceedings of the Fifth International Conference on Machine Learning.*

Thagard, P. (1988). The conceptual structure of the chemical revolution. Unpublished manuscript.

Zytkow, J. M., & Simon, H. A. (1986). A theory of historical discovery: The construction of componential models. *Machine Learning 1*, 107–136.

called green-solid (K A → H G), plus another observation (G W → P A W)) leads to this effect.

TEXT COMPREHENSION : MACROSTRUCTURE AND FRAME

J.P. ROSSI
Université de Poitiers
Laboratoire de Psychologie du Langage, CNRS U.A. 666

In the last 15 years, cognitive research on text comprehension has focused either on the description of the propositional content of texts (analysis of relationships between propositions or concepts), or the study of a pre-existing pattern stored in long term memory alternatively called : "frame" (Minsky, 1975; Frederiksen, 1986), "script" (Schank and Abelson, 1977), "scenario" (Sanford and Garrod, 1981) or "schema" (Rumelhart and Ortony, 1977).

These different terms correspond to varieties of schemas. The text schema is a semantic network which describes the common components of classes of texts and their interrelationships. These are composed of nodes and links. In Mandler and Johnson's (1977) story grammar theory the nodes are Beginning (B), Reaction (R), Goal (G), Attempt (A), Outcome (O) and Ending (E). In the Beginning (B) the protagonists are faced with a problem, they React (R), formulate a Goal (G), produce an Outcome (O). The Ending (E) is the conclusion. For the partisans of story grammars this typical network is progressively internalized by individuals through repeated exposure (hearing and reading) to stories (Denhiere, 1986). Few studies have been conducted on scientific text schema, with the exception of Frederiksen (1986), who introduced the notion of "problem frame". The principal nodes of this frame are: problem states, procedures, result states and goal. The problem frame will be employed in the experiment described below. Frames guide comprehension and govern the recovery of information stored in memory when the individual must recall or summarize a story, (Denhiere, 1986).

Current research aims at constituting text grammars and evaluating the role of schemas in text encoding, text comprehension and recall. Cross-cultural invariance of narrative structure has been attested to for literate and nonliterate societies as well (Mandler, Scribner, Cole & Deforest,1980). People recall stories through schemas (Glenn, 1978).

Comprehension is facilitated if the text fits in the schema (Kintsch and Yarbrough, 1982). In educational studies, Rahman and Bisanz (1986) showed that in reconstruction and recall tasks, both good and poor readers use a story schema when the story followed the canonical format but poor readers' story schema "was either not as well developed or as efficiently used as good readers'". Cuing poor readers to use a story schema does not facilitate their recall. These data are not informative as to how schemas intervene in reading comprehension because a good reader is an individual who constructs a good macrostructure and a macrostructure is good if it reflects the text schema. To evaluate the function of schema in text comprehension it is necessary to provide a description of what is theoretically happening as one reads. This requires opposing two basic modes of information processing: a top-down model and a bottom-up model. In the first mode, the top-down model, the reader activates a schema and searches for information to fit in each node (Adams & Colins, 1980). The bottom-up processing, works in the opposite direction: in the absence of pre-existing schemas, the cognitive system constructs a macrostucture to reflect the text schema.

The present experiments were designed to validate mode of information processing. If the activation of schema facilitates text comprehension, performance in comprehension and summarization tasks should be improved if the subject is given information on the schema. In contrast, if comprehension is independent of schema activation, information on the schema should prove inefficient, and only macrostructure information should have a positive effect. Similarly, if the schema is purely a means for constructing the macrostructure, the activation of an adequate schema should improve performance, but to a lesser extent than macrostructure information alone.

METHOD

Material

Two extracts from 5th grade school textbooks of were selected. The first, entitled "The Invention of The Printing Press" was composed of 208 words, 22 lines, 10 sentences, 126 propositions (Turner and Green,1978) and 5 paragraphs. It was used for practice, and as a pretest for matching subjects.

The second text used in the 5 conditions dealt with "How to Measure the Distance between Stars". It was made up of 237 words, 25 lines, 14 sentences and 142 propositions. The text was divided into 5 paragraphs : the first presented the problem, the second gave an explanation, and each of the three remaining paragraphs suggested a solution and provided an example.

The macrostructure appeared as the last sentence of each paragraph. Both texts were typed on sheets of paper.

A five-item questionnaire on macrostructure constituents was devised for each text. The first question dealt with the text topic (first paragraph), e.g. "What is the text about?". The second item dealt with the second paragraph and so on. Subjects were asked to respond in writing in the space provided below each question. A blank sheet was furnished to write a summary.

PROCEDURE AND DESIGN

Five conditions were tested :

Group 1. texts with no annotation (control group)

Group 2. texts with macrostructure sentences underlined

Group 3. texts where the frame (headings corresponding to the text structure) were indicated in list form inserted between the title and the beginning of the text (problem, explanation of problem, first solution and first example, second example, third example..)

Group 4. the frame headings were indicated in the margin next to each paragraph

Group 5. frame information identical to Group 4, plus macrostructure sentences underlined as in Group 2.

Hypotheses as to the role of the frame and the macrostructure were tested by comparing performances across groups. If the frame alone is important, then $G1=G2<G3<G4=G5$, if the frame has no effect, then $G1=G3=G4<G2=G5$. In contrast, if the schema guides the construction of the macrostructure, then $G1<G3<G4<G2<G5$.

The subjects read the text, wrote a summary and answered the questionnaire. They were informed that each text was scored on the basis of 20 points and that they would be allotted 25 minutes. Subjects were allowed

to have the text in front of them during the entire experiment and could consult it freely when writing the summary and answering questions.

The experiment was divided into two phases. In phase 1, all the subjects read the text entitled "The Invention of the Printing Press", wrote the summary and answered the questions. These results were then used for matching subjects who were assigned to each of the five conditions. Phase 2 was identical to Phase 1 except that the text used was "How to measure the distance between stars" and the information provided to the subjects varied across conditions. The two phases were administered at a two-week interval in the same school on Mondays and Tuesdays.

Subjects

The sample was composed of 100 children in the last years of French elementary school (5th grade). The mean age was 10,1 years +/- 6 months. The subjects were attending 4 schools which catered to similar socioeconomic backgrounds. As a function of scores on the first text, the subjects were assigned to one of the 5 conditions, to form matched groups.

Scoring

A grid indicating the correct answers to each item and the score was constructed by two judges. Each question was scored on the basis of a total of 4 points, yielding a maximum score for each text of 20 points. Scoring of the summaries was obtained by comparing subjects' responses with a model summary produced by two judges. Responses were weighted so that each of the five parts of the text had the same number of points. Maximum score was 20.

Responses to the questionnaire and the summaries were scored independently by two judges. Disagreements were easily resolved through discussion.

On the basis of performance on text 1, five homogeneous matched groups were formed. The means for these groups appear in Table 1. No significant differences were observed, and the Fs were all less than 1.

RESULTS

The means for the questionnaire and the summary for the 5 groups appear in Table 1.

TABLE 1. Means and standard deviations for scores of the 5 groups of subjects

Groups		G1	G2	G3	G4	G5
Quest.	M	9.05	12.20	11.00	11.35	11.95
	S.D.	4.70	4.26	3.93	3.54	5.81
Summary	M	9.35	13.60	11.25	11.15	12.35
	S.D.	5.52	5.51	5.28	4.34	3.60

Questionnaire Results

The control group (G1) performed significantly worse ($F(1,95)=5.19$ $p<.01$) than the other groups. The four experimental groups (G2,G3,G4,G5) exhibited equivalent scores ($F<1$). When the problem-frame was indicated in the margin, performance was not better than when the problem-frame was listed at the start of the text : $F<1$ for group 3 vs Group 4. Similarly, performance did not differ when the problem-frame information was added to the macrostructure (group G2 vs G5, $F<1$) or when macrostructure information was added to the problem-frame (G5 vs G3G4 $F<1$). Regardless of the type of annotation, problem-frame information was as efficient in improving performance as underlining the macrostructure (G2 vs G3 G3, $F<1$).

Summary Results

The summary means do not differ from those obtained on the questionnaire ($F<1$). The observed effects were identical : positive impact of aid in the form of annotation as compared to the control group text ($F(1,95) = 4.96$ p.<015); no significant differences between the experimental groups ($F(3,76) = 1.16$, ns).

The group x type of task interaction did not reach significance ($F<1$). The scoring system for the summary obviously generates this result.

The analyses presented above were based on group data. It is likely however that the efficiency of aid will vary according to the type of subject. Arguably annotation would be of little help to subjects who already implement the type of processes indicated, but on the contrary would be highly efficient in subjects who do not know how to proceed. Thus readers who scored low on the practice text were assumed not to have mastered adequate processing techniques in contrast to high scorers, who were assumed

to use efficient strategies. On the basis of the score for comprehension questions on the practice text, 2 groups of subjets were formed: Good Comprehenders (G.C.) whose comprehension scores were =/> 12; Poor Comprehenders (P.C.) whose scores were < 10. Summary scores were handled in the same way: scores for Good Summarizers (G.S.) were =/> 12; Poor Summarizers (P.S.) scored under 10. The number of subjects and the mean performances for these two groups appear in Tables 2 and 3.

The two groups differ significantly ($F(1,80) = 29.70$; $p .<.001$).

All the aids improve performance significantly in Poor Comprehenders : $F(1,35) = 5.26$, $p.< .02$ but the means for the four experimental groups (G2,G3,G4,G5) do not differ significantly ($F<1$).

In contrast, the performance of G.C. were significantly improved when the macrostructure was underlined. Only this condition, when compared to those where the macrostructure was not underlined, reaches significance ($F(1,27) = 13.15$, $p.=.001$).

TABLE 2. Questionnaire results for good and poor comprehenders

Groups		G1	G2	G3	G4	G5
PC	n	9	8	7	6	10
	M	5.44	10.62	10.28	7.66	9.40
	S.D.	3.71	3.58	4.99	3.26	6.86
GC	n	7	9	10	11	8
	M	12.28	15.66	10.50	13.36	15.37
	S.D.	3.40	3.24	3.27	2.27	2.38

TABLE 3. Summary results for good and poor summarizers.

Groups		G1	G2	G3	G4	G5
PS	n	10	10	11	9	10
	M	5.60	10.60	9.81	9.33	10.20
	S.D.	4.27	5.08	5.47	4.06	3.01
GS	n	7	8	6	9	7
	M	12.85	18.25	13.50	13.37	14.57
	S.D.	4.56	2.05	3.67	4.65	3.45

The summary results (see Table 3) exhibit similar trends. All the aids significantly improve performance of Poor Summarizers whereas only the underlining of the macrostructure is significantly efficient for Good Summarizers ($F(1,27) = 5.07$, p.03).

CONCLUSION

Overall, the findings indicate that problem-frame knowledge is as useful as macrostructure knowledge for text comprehension and summarizing. The frame thus appears to have an effect similar to the macrostructure on text processing.The differences in efficiency of these two types of aids only appear when the subjects are classified as good or poor comprehenders/ summarizers.

All the aids tested in this study proved to be efficient for poor comprehenders and summarizers. Frame knowledge and the macrostructure are equally efficient for Poor Comprehenders but their processing strategies are apparently inefficient. All aids have the effect of improving performance and none interfered with their mode of processing. However the improvement observed in these subjects was not sufficient to bring them up to the level of the good comprehenders who received no aid. This seems to suggest that poor comprehenders/ summarizers use the information provided to them in the form of aids, but do not integrate this information into their processing strategies.

Only macrostructure information improves performance in Good Comprehenders and Summarizers whereas problem-frame information has no effect. There are three possible explanations for this. First of all, frame information may not be necessary to comprehension. Secondly, frame information was redundant because these subjects already had a problem-frame available. Thirdly, the frame provided was not sufficient for the construction of the macrostructure. This could have been because the problem-frame was inappropriate or because the operations necessitated by the activation of the frame were complex.

BIBLIOGRAPHIE

Adam, J-M. (1986). - Typologies énonciatives et séquentielles: quelle(s) approche(s) de la textualité? Communication au congres "Texts and texts processing" , Poitiers.

Bartlett, F.C. (1932). - Remembering: a study in experimental and social psychology , Cambridge University Press.

Denhière, G. (1984). - Il était une fois... Compréhension et souvenir de réçits , Presse Universitaires de Lille, Lille.

Denhière, G. (1986). - Connaissances générales et connaissances linguistiques dans la production de textes, Communication au 3° colloque internationnal de Didactique du français , Namur.

Frederiksen, C.H. (1986). - Constructing and manipulatoring multi-level semantic representation in comprehending and producing text. Communication au congres "Texts and texts processing" , Poitiers.

Glenn, C. (1978). - The role of episodic structure and of story length in children's recall of simple stories. Journal of Verbal Learning and Verbal Behavior , 17 , 229-247.

Kintsch, W., van Dijk, T.A. (1978) - Towards a model of text comprehension and production. Psychological Review , 85 , 364-394.

Kintsch, W. & Yarbrough, J.C. (1982). - Role of rhetoric structure in text comprehension. Journal of Educational Psychology , 74 , (6), 828-834.

Mandler J.M. & Johnson, N.S. (1977). - Remembrance of things parsed: story structure and recall. Cognitive Psychology , 9 , 111-151.

Mandler J.M. & Scribner S., Cole M. & Deforest M. (1980). - Cross-cultural invariance in story recall. Child Development , 51 , 19-26.

Minsky, M. (1975). - A framework for representing knowledge. In P.H. Winston (Ed.), The psychology of computer vision . New-York: Mc Graw-Hill.

Rahman, T., Bisanz, G.L. (1986). - Reading ability and use of a story schema in recalling and reconstructing information, Journal of Educational Psychology , 78 (5), 323-334.

Rumelhart, D.E., Ortony, A. (1977). - Representation of knowledge, in R.C. Anderson, R.C. Spiro, & W.E. Montague (eds), Schooling and acquisition of knowledge. Hillsdale, New-York: Lawrence Erlbaum Associates.

Sanford, A.J., Garrod, S. (1981). - Understanding written language . Chichester: Wiley.

Schank, R., Abelson, R. (1977). - Scripts, plans, goals and understanding . Hillsdale, New-York: Lawrence Erlbaum Associates.

THE STRUCTURE OF SOCIAL MIND: EMPIRICAL PATTERNS OF LARGE-SCALE KNOWLEDGE ORGANIZATION

David L. Sallach
Department of Computer Science
University of Arkansas

The social model of mind regards intelligence as the result of the interaction of cognitive or subcognitive agents. In recent years, cognitive science research has found the concept of social mind provides a promising model for: 1) the explanation of brain damage (Gardner, 1974; 1983), 2) the design of expert systems (Lee, 1985), and 3) a general model of memory (Minsky, 1981;1985). The premise of the present discussion is that the structure of social mind has research implications for cognitive science that have yet to be explored.

The study of biological systems has yielded genetic algorithms that provide useful insights into the design of classifier systems (Holland, 1986).[1] Like biological systems, social systems are naturally occurring phenomena.[2] Thus, identification of the principles and organization of social entities also provides insights which may facilitate the construction of computational cognitive systems.

Social entities can be seen as providing a functional architecture through which mind can be instantiated. Accordingly, the study of cognitive systems will benefit by including the social system as a physical instantiation of mind, along with the computer and the human brain. For some cognitive science research questions, social systems are likely to provide a more useful source of data . Further, the models of social mind used by cognitive researchers are likely to be enriched by an investigation of empirical forms of social organization.

FUNCTIONALISM AND SOCIAL ENTITIES

A major side-effect of the modern computer revolution is the growing recognition among philosophers and cognitive scientists that thinking and other mental functions may be instantiated using diverse mechanisms (Dennett, 1981; Loar, 1981; Churchland, 1984; Gardner, 1985). Cognitive research increasingly relies upon both psychological experiments and computer simulation (cf., Kintsch *et al.*, 1984). Cognitive psychology provides insight and hypotheses to AI researchers (Hinton and Anderson,1981), while computer modeling enriches cognitive psychology (Johnson-Laird, 1983).

One result of cognitive research considering comparative mental mechanisms is the abstraction of function, architecture and representation from physical implementation. Development of the functional perspective in cognitive science has led to the identification of three distinct levels of analysis: the physical level, the functional level and the knowledge or representational level. "The three levels are tied together in an instantiation hierarchy, with each level instantanting the one above" (Pylyshyn, 1984, p. 132).

Figure 1 illustrates the relationship between cognitive science and the disciplines upon which it is based. The highest level Newell calls the knowledge level, Pylyshyn calls the

---

[1]Not surprisingly, genetic algorithms have been shown to provide especially effective models of biological phenomena (Farmer, Packard and Perelson, 1985).

[2]Social systems differ from the simpler biological systems in that their actions are frequently cognitively penetrable; see Pylyshyn (1984, pp. 130-145) for a discussion of penetrability.

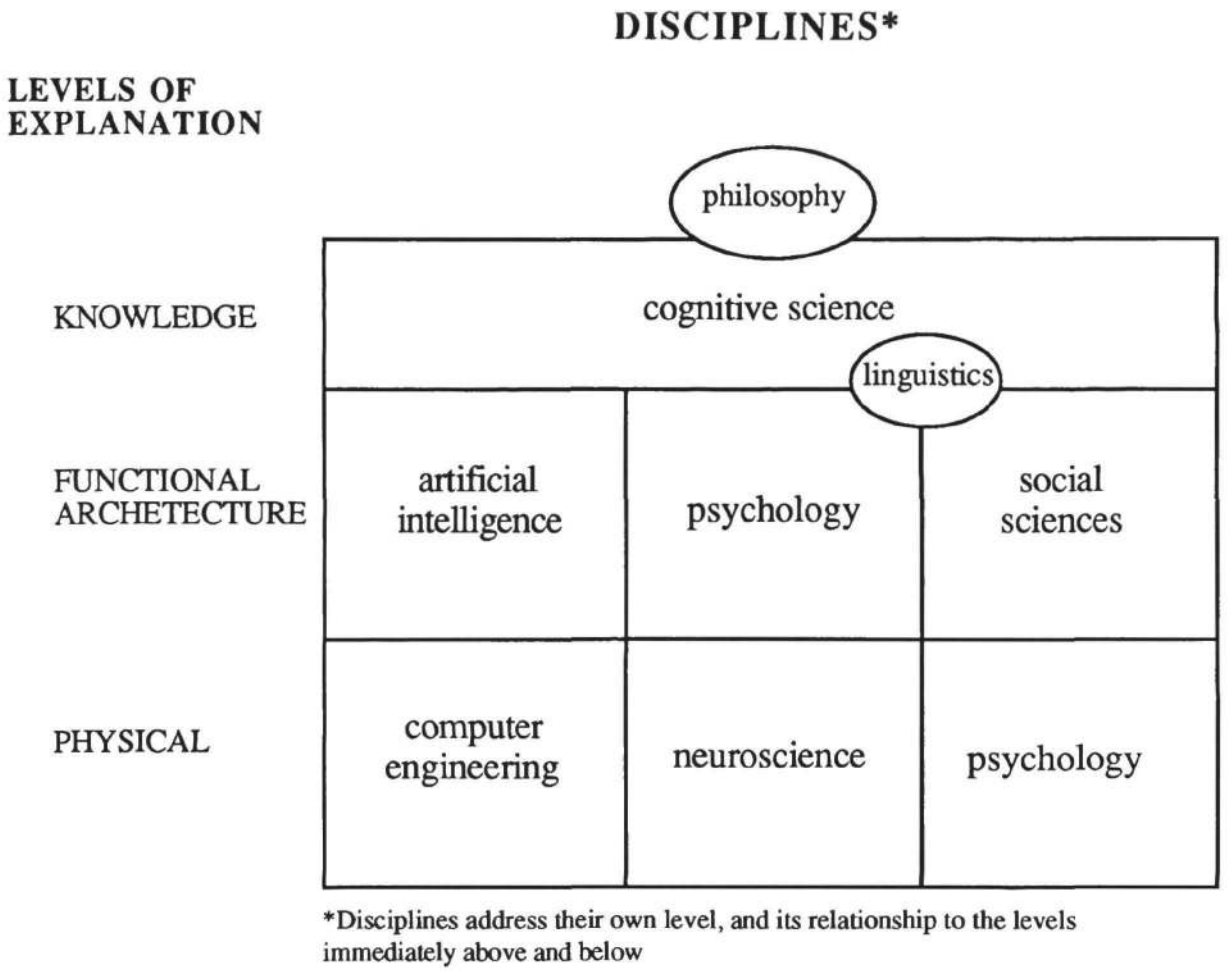


Figure 1. Cognitive Science Disciplines and Levels of Explanation

semantic level and philosophy references as the intentional level. It is the domain of cognitive science.[3]

The knowledge level may be instantiated using diverse functional architectures. The relationship between the knowledge level and prospective functional architectures is of interest to cognitive science, and to disciplines that study the architectures available through a specific type of physical system. For computers, the discipline is artificial intelligence; for *homo sapiens*, the discipline is psychology. By extension, the relationship between the knowledge level and the functional organization of social entities falls within the domain of the social sciences.

A second, more concrete, focus concerns the types of functional architecture that are available for a specific type of physical system. For computers, this relationship is studied by artificial intelligence and computer engineering (which together constitute a major portion of

[3]Figure 1 accurately depicts philosophy as addressing levels of analysis more abstract than that of cognitive science (e.g., ontology, epistemology), while philosophy of mind overlaps and interacts with cognitive theory. Figure 1 would be slightly more accurate if it showed the overlap and interaction between philosophy and linguistics.

computer science). For the human individual, this relationship is studied by neuroscience and physiological psychology. For social entities, the relationship between the physical system, and its possible and probable forms of functional organization is studied by the social and psychological sciences.

When applied to social phenomena, the functional theory of mind suggests that mental representations and processes may be attributes of social, as will as psychological and computational, entities. Dennett (1981) recognizes that the very virtues of functionalism[4] permit a functionalist theory to be instantiated by suprahuman organizations for which it may seem counterintuitive to say they have minds of their own. Intuition, however, is not a reliable inidicator. As Dennett further observes, "Inside your skull it is also all darkness, and whatever processes occur in your grey matter occur unperceived and unperceiving."

Even critics of functional theory have (skeptically) suggested that, under functional theory, social entities (e.g., the people of China) might form "a giant brain" (Churchland, 1985, p. 39). While it is more accurate to say "a giant mind", the people of China are hardly the only social entity that acts based upon shared representations and collective goals.

Over the past century, theorists have attributed various levels of 'reality' to social phenomena. The perspective developed in the present discussion is that of social realism. From its earliest articulation, social realism has been anti-reductionist in nature (Durkheim, 1964). In general, social realism holds that: 1) social entities are just as real as psychological entities, but that 2) both are abstract, analytical units, and 3) social phenomena must be explained in terms of a social level level of analysis, not reduced to a psychological one (cf., Warriner, 1956).

Social realism, as described above, allows and requires two caveats: 1) like other phenomena, the behavior of social entities is constrained by the nature of its compositional elements, whether psychological, biological, or physical, and 2) social patterns may be partially or wholly caused by forces which are better conceptualized at a higher, more abstract level of analysis. The latter caveat acknowledges that, insofar as the scientific comprehension of cognitive systems in general grows, social entities may be seen as providing one type of physical system through which a range of functional architectures are implemented. The subsequent review of research is based upon this perspective.

Recognition of the mental dimensions of social phenomena will benefit cognitive science research. Social entities are spatially much larger than the brain, and their communication rates are much slower and (frequently) more observable. It is not clear, however, that the cognitive process of social entities is any less complex than that of the brain. Cognitivists enjoy a unique vantage point on these large-scale cognitive processes: we are on the inside looking out. Incorporating this unique perspective into our research programs is likely to widen the range of cognitive issues that can be addressed.

THE BELIEF SYSTEMS OF MASS PUBLICS

As a preliminary example of how empirical research on the social organization of cognition may provide useful insights into the construction of computational models, consider nature of mass belief systems.[5] It has long been recognized that opinion formation is a multi-stage process in which opinion leaders form their views based upon public sources of information, and then influence the attitudes of a periphery of opinion followers (Berelson, et al., 1954). This type of layered structure gives support to Minsky's (1981;1985) K-line model in which there is a division of labor between local agents and agents that perform more generalized pattern recognition.

---

[4]To wit, " . . .abstractness and hence neutrality with regard to to what could 'realize' the functions deemed essential to sentient or intentional systems" (Dennett, 1981, p. 153).

[5]Although the research to be reviewed pertains to the organization of political knowledge mass publics, there is no inherent reason to believe that the structural patterns are restriced to that domain.

Subsequent studies suggest further implications. Converse (1964), in a highly influential study, concluded that the mass public manifests belief systems organized at several identifiable layers of organization, ranging from a coherent organization of information at a high level of abstraction, through interest group identification, to those whose information is very specific, and not coherent or integrated. In Converse's words:

> Moving from the top to the bottom of this information dimension the character of the objects that are central in a belief system change. These objects shift from the remote, generic and abstract to the increasingly simple concrete, or 'close to him'.

Converse's findings of differential levels of abstraction and conceptual integration in the belief systems of mass publics have been elaborated by subsequent research. Neuman (1981) has identified two complimentary dimensions of political thought: conceptual differentiation and conceptual integration. Differentiation refers to the number of discrete, concrete elements of information utilized by the actor. Conceptual integration involves the use of abstract concepts to structure discrete elements of information. Neuman posits a spiraling pattern of growing conceptual sophistication in which new elements of information are acquired (differentiation) and then abstracted into a manageable order (integration). The structure of belief then guides and constrains subsequent information acquisition.

Brady and Sniderman (1985) have isolated another aspect of mass belief systems that may be relevant to more generalized models of cognition. Members of mass publics are able to make highly accurate estimates of the political beliefs of strategic groups, and how members of such groups are likely to line up on key issues. This accomplishment is a puzzle since most members of mass publics demonstrate a low level of abstraction and little concrete information. The mechanism by which such estimates are made apears to involve an affective heuristic that serves as an intellectual shortcut. Specifically, respondants appear to combine their own beliefs with their affective response to strategic groups and generate an impressively accurate map of the political landscape. Other research suggests heuristics based upon multiple affective dimensions, for example, evaluation, potency and activity (Osgood, 1962; Heise, 1979).

This cursory review of research on mass belief systems suggests four components that might usefully be incorporated into models of mind based upon social organization. The first component differentiation of agents based upon their location in a knowledge hierarchy varying in level of abstraction and information span. A second component is the use of an affectivity calculus to enhance the application of agent knowledge, extending it at lower levels of the knowledge hierarchy and accentuating it at higher levels. The final component is the dynamic evolution of agents, through a spiral process, from a minimum information span at a low level of abstraction to a broader information span at successively higher levels of abstraction. Agent evolution implies the creation of new agents and the selective retention of mature agents (cf., Holland, *et al.*, 1986).

THE STRUCTURE OF SOCIAL MIND

The social organization of knowledge implies, in addition to agents (individually and collectively), a number of intermediate social entities. Social organizations are composed of smaller organizations which are ultimately composed of groups of agents that interact directly. Thus, the structure of social organizations is defined recursively to an arbitrary depth, further supporting the use of agents that possess extensive modeling capabilities.[6] In terms of functional architecture, social organizations are massively parallel systems composed of intelligent agents.

---

[6] The view developed here posits complex cognitive systems being composed, recursively, of complex cognitive systems. Such structures parallel the universality of scale identified in research on nonlinear dynamics in multiple domains (cf., Cvitanovic, 1986).

The information processing capacity of component agents is not infinite, but it is of sufficient depth that system designers may utilize significant levels of agent complexity.

The computational requirements implied by a system with a recursively nested structure composed of massively parallel agents, each having extensive modeling capabilites, are constrained by a shared definition and utilization of mental models among the multiple agents. Specifically, agents are not discrete entities, but nodes that have access to a larger knowledge network of which they are a part. Non-specialist subgroups and individual agents may thus be conceived as non-autonomous, maintaining simpler local models of objects and object types that draw upon the larger network model(s).

The simple cognitive models allow routine information processing, and serve as associative pointers to the more detailed information available as needed from the higher level agents of which they are a part. The integration of related models at multiple organizational levels is not a result but a task, an ongoing process that is achieved only partially (Schutz, 1967). In social systems, large salient model discrepencies lay the basis for conflict and disorderly change.

A major feature of social mind, emphasized by most theorists who invoke the model, is the use of a functional division of labor. The existence of specialists and generalists suggests a diffuse form of organization connected by another type of (hierarchical or heterarchical) structure. This second (control) structure determines the flow of information and control across multiple levels. The location of an agent within the control structure defines a vertical dimension of social structure. The advantages and disadvantages of alternative control structures constitutes a major topic of research for the social model of mind.

Empirical social systems suggest another type of differentiation: grouping across physical space. Spatial grouping provides communication opportunities and obstacles which, in social systems, result in complex patterns of ethnicity, language, religion, nationality and tradition. Spatial differentiation allows alternative organization and models to develop. Accordingly, it is a potential source of both experimentation and poor social integration. Control structure research might reasonably focus on the identification of structures which benefit from the former while minimizing the latter.

Thus, social models are constituted by a minimum of three dimensions of structure: division of labor, spatial dispersion and stratification of control (cf., Blau, 1977). Each agent, whether individual or social, is defined in terms of each dimension. Models of relevant objects are distributed across the matrix. The composite model of an object may be regarded as a conceptual prototype (cf., Rosch, 1978, ; Lakoff, 1987; Sallach, 1988).

The relationship between the composite model of an object, and the model used by each agent may be regarded as analagous to the multiple levels of database design, where the user view contains the information needed by a specific type of user, while the conceptual view integrates the user views of an enterprise into a single schema (cf., Flavin, 1981). However, in the database analogy, schema definition is performed by the database administrator. In the social model of mind, the integration of the models of multiple agents must be a self-organizing process.

THE REPRESENTATION OF SHARED KNOWLEDGE

The previous section describes a common knowledge network, where the detail of local representation is determined by the functional requirements of the position, and simple local models point to more generalized, or specialized repositories of information. A structure of this type is both efficient (in limiting information redundancy) and problematic (in coordinating the information needs of the agent with the resources of specialist nodes and of the network as a whole). The problem of representing such a knowledge network is largely a problem of the efficient integration of inconsistent knowledge. What is needed is the equivalent of a self-organizing schema for a semantic data model.

From the mass public, through intermediate organizations, to the individual agent, there are multiple levels of information integration to be reconciled in a social model of mind. The general pattern is that agents that are high in the social knowledge system are likely to have highly

integrated information structures. Other agents composing the social knowledge system will vary in the extent of information integration, and in the span of locally available information.

The structure of mass belief systems suggests a strategy for knowledge representation for the social model of mind. Within the social knowledge system as a whole, there will be islands of information that are comprehensive and highly integrated. For these nuclei, coherent schemata are generated. Entities whose information span is narower but consistent, are defined in terms of a broader schema (the equivalent of a database 'view'). Entities whose information is coherent but organized according to different abstractions are represented as alternate schemata.[7] Finally, entities and agents that fail to manifest a minimal level of integration are defined in terms of an affective heuristic.This strategy for representation of knowledge shared across a network is neutral as to whether the core schemata are defined in terms of individual agents or organization of agents. If, however, the level of integration is equivalent, the larger social entity would be the unit of preference.

The social knowledge system as a whole would not be presumed to manifest a high level of integration. Rather, knowledge system integration would be used to: 1) to provide a heuristic for the evaluation of alternate schema organization, and 2) as a variable property which constitutes an appropriate subject of investigation.

CONCLUSION

Social models of mind have been useful because of their ability to represent a functional division of labor among semi-autonomous agents. The present discussion maintains that mind may be usefully viewed as instantiated by social systems. This perspectives suggests that the empirical investigation of social systems can provide useful insights into the social organization of cognitive systems.

The study of belief systems in mass publics illustrated three principles of the society of mind: the organization of knowledge along hierarchies of conceptual differentiation and integration, 2) the simulation of knowledge among low-level agents by the use of affective heuristics, and 3) dynamic learning through the evolution of agents. The structure of social mind is defined by a minimum of three dimensions (function, control and physical location), which suggests that system knowledge is distributed across a network of agents and organizations. The use of self-organizing schemata as a form of non-redundant knowledge representation was explored.

REFERENCES

Berelson, B.R., Lazarsfeld, P. and McPhee, W.N. (1954). *Voting: A Study of Opinion Formation in a Presidential Campaign*. Chicago: University of Chicago Press.

Blau, P.M. (1977). *Inequality and Heterogeneity: A Primitive Theory of Social Structure*. New York: Free Press.

Brady, H.E. & Sniderman, P.M. (1985). Attitude attribution: A group basis for political reasoning. *American Political Science Review*, 79, 1061-1078.

Carley, K. (1986). An approach for relating social structure to cognitive structure. *Journal of Mathematical Sociology*, 12, 137-189.

Carroll, J. and Payne, J.W. (Eds.). (1976). *Cognition and Social Behavior*. Hillsdale, NJ: Lawrence Erlbaum.

Churchland, P.M. (1984). *Matter and Consciousness*. Cambridge: MIT Press.

Converse, P.E. (1964). The nature of belief systems in mass publics. In D. Apter (Ed.), *Ideology and Discontent*. New York: Free Press.

Cvitanovic, P. (1986). *Universality in Chaos*. Bristol, UK: Adam Hilger.

[7]Several authors have developed procedures for assessing cognitive integration (cf., Schoder, Driver & Streufert, 1967; Carley, 1986; Smolensky, 1986).

Dennett, D.C. (1981). Toward a cognitive theory of consciousness. In *Brainstorms: Philosophical Essays on Mind and Psychology.* Cambridge, MA: MIT Press

Durkheim, E. (1964). *The Rules of Sociological Method.* New York: Free Press.

Farmer, J.D., Packard, N.H. and Perelson, A.S. (1985). The immune system, adaptation and machine learning. *Technical Report LA-UR-85-3151.* Los Alamos National Laboratory.

Flavin, M. (1981). *Fundamental Concepts of Information Modeling.* Englewood Cliffs, NJ: Yourdon.

Gardner, H. (1974). *The Shattered Mind.* New York: Vintage.

Gardner, H. (1983). *Frames of Mind: The Theory of Multiple Intelligences.* New York: Basic Books.

Gardner, H. (1985). *The Mind's New Science: A History of the Cognitive Revolution.* New York: Basic Books.

Heise, D.R. (1979). *Understanding Events: Affect and the Construction of Social Action.* New York: Cambridge.

Hinton, G.E. and Anderson, J.A. (Eds.). (1981). *Parallel Models of Associative Memory.* Hillsdale, NJ: Lawrence Erlbaum.

Holland, J.H. (1986). Escaping brittleness: The possibilities of general purpose machine learning algorithms applied to parallel rule-based systems. In R.S. Michalski, J.G. Carbonell and T.M. Mitchell (Eds.), *Machine Learning: An Artificial Intelligence Approach,* vol. 2. Los Altos, CA, Kaufmann.

Holland, J.H., Holyoak, K.J., Nisbett, R.E., and Thagard, P.R. (1986). *Induction: Processes of Inference, Learning and Discovery.* Cambridge: MIT Press.

Johnson-Laird, P. N. (1983). *Mental Models: Towards a Cognitive Science of Language, Inference and Consciousness.* Cambridge: Harvard University Press.

Kintsch, W., Miller, J.R. and Polson, P.G. (Eds.). (1984). *Method and tactics in Cognitive Science.* Hillsdale, NJ: Lawrence Erlbaum.

Lakoff, G. (1987). *Women, Fire and Dangerous Things: What Categories Reveal about the Mind.* Chicago: University of Chicago Press.

Lee, R.M. (1985). Bureaucracy as artificial intelligence. In L. B. Methlie & R.H. Sprague (Eds.), *Knowledge Representation for Decision Support Systems.* New York: North-Holland.

Loar, B. (1981). *Mind and Meaning.* New York: Cambridge University Press.

Minsky, M. (1981). K-Lines: A Theory of memory. In D.A. Norman (Ed.), *Perspectives on Cognitive Science.* Hillsdale, NJ: Lawrence Erlbaum.

Minsky, M. (1985). *The Society of Mind.* New York: Simon and Schuster.

Neumann, W.R. (1981). Differentiation and integration: Two dimensions of political thinking. *American Journal of Sociology* 86, 1236-1268.

Osgood, C.E. (1962). Studies of the generality of affective meaning systems. *American Psychologist* 17, 10-28.

Pylyshyn, Z.W. (1984). *Computation and Cognition: Toward a Foundation for Cognitive Science.* Cambridge: MIT Press.

Rosch, E. (1978). Principles of categorization. In E. Rosch & B.B. Lloyd (Eds.), *Cognition and Categorization.* Hillsdale, NJ: Lawrence Erlbaum.

Sallach, D.L. (1988). The representation and use of composite prototypes in problem-solving tasks. *Technical Report CSAS-TR 88-02.* Fayetteville: University of Arkansas.

Schroder, H.M., Driver, M.J., and Streugert S. (1967). *Human Information Processing.* New York: Holt, Rinehart and Winston.

Schutz, A. (1967). *The Phenomenology of the Social World.* Evanston: Northwestern.

Smolensky, P. (1986). Information processing in dynamical systems: Foundations of Harmony Theory. In D.E. Rumelhart & J.L. McClelland (Eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition,* Volume 1. Cambridge: MIT Press.

Warriner, C. (1956). Groups are real: A reaffirmation.*American Sociological Review,* 21,549-54.

A Model of Meter Perception in Music

Benjamin O. Miller, Don L. Scarborough and Jacqueline A. Jones
Brooklyn College of the City University of New York

ABSTRACT

A fundamental problem in music cognition is the question of how the listener extracts the music's temporal organization. We describe a model, implemented as a computer simulation, that constructs a hierarchical representation of metric structure that conforms to the requirements of Lerdahl & Jackendoff's (1983) generative theory. The model integrates bottom-up processing of score data with top-down processes that generate predictions of temporal structure, and with rules of organization that correspond to musical intuition. Several examples of the program's output are used to illustrate these processes.

INTRODUCTION

People perceive patterns in temporal events even in the absence of any physical cues. For example, we hear a sequence of identical equally spaced tones as being grouped by twos or possibly threes (Fraisse, 1982), with the first tone of each group accented. When physical cues, such as accents or pauses, are present, perceptual grouping is that much more robust. A result of such perceptual processes is meter. A meter specifies a perceived pulse that marks off equal intervals in time. Pulses tend to be grouped, with the first of each group heard as accented. Within groups, strong and weak pulses alternate in a way that reflects hierarchical organization, as illustrated below. In the dot notation introduced by Lerdahl & Jackendoff (1983), 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. Pulses with dots at more than one level are perceived as stressed relative to others, and 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**.

```
          Pulse   1  2  3  4  5  6  7  8  9  10  etc.
                1 .  .  .  .  .  .  .  .  .  .
Metric level    2 .     .     .     .     .
                3 .           .           .
```

Rhythm emerges from the interaction of metric structure with auditory events. For example, syncopated rhythms occur when perceived musical accents are heard as occurring at unaccented positions in the metric structure. Thus, musical events are heard within a framework established by the metric structure. On the other hand, the music itself must also guide the listener in establishing a metric framework for the interpretation of the music. We are interested in how a listener discovers a metric structure.

We have been developing a model of meter and rhythm perception based on Lerdahl & Jackendoff's (1983) *Generative Theory of Tonal Music*, or GTTM. GTTM attempts to formalize the intuitions of a human listener regarding classical Western tonal music. The theory does this by means of four stages of analysis, each embodied in two sets of rules. Well-formedness rules (WFRs)

are analogous to grammatical rules, in that they specify legal structures within a stage, and preference rules (PRs) correspond to laws of perceptual organization. We are modeling, as a computer simulation, the first stage, metric analysis, which yields a hierarchical representation of metric structure that conforms to traditional intuitions about meter and accent.

In previous work, there have been five very different computer simulation models of meter and rhythm perception. Differences among the models have to do with: the method of analysis--that is, note-by-note or all-at-once (i.e. the entire score is available in constructing the analysis); the type of data used (i.e. time intervals only or time and pitch); their ability to deal with anacrucis (initial unaccented notes), syncopation, and ambiguity; and the nature of their output. All simulations are limited to single voice input.

Simon's (1968) all-at-once LISTENER program groups the note durations it uses as input into units of equal duration. From this it extracts repeating rhythmic phrases that may span several measures. LISTENER does not distinguish between meter and rhythm. Longuet-Higgins & Steedman (1971), developed a note-by-note parser that adopts the first note-value as the basic metric unit and adjusts it based on successive note values. It cannot handle passages of notes of equal duration. A later program by Steedman (1977), using the output of the Longuet-Higgins & Steedman program, makes a second pass through the score, considering not only note-values but also melodic repetition, assuming that size and separation of a figure and its repeat reflect the metric structure.

A fourth program (Longuet-Higgins & Lee, 1982) returns to a time-only orientation as well as a note-by-note approach. The program takes a list of note values (onset to onset) as input, and uses four productions to generate a metric unit and the location of bar lines as output. This program is a major improvement over earlier programs. It handles syncopation and anacrucis, and it is also hierarchical, identifying more than one metric level as it progressively expands the metric unit. However, since it analyzes until the metric unit reaches a maximum size of one whole note and then stops, it cannot detect changes in meter in the middle of a piece, nor does it produce an analysis of the whole piece.

Grid theory (Povel, 1984) is a different approach to time-only all-at-once analysis. A grid is analogous to a metronome that ticks at a uniform rate. If a metronome ticks while music plays, we can talk about the fit between the two in terms of how many ticks coincide with note onsets (**hits**; the more the better) and how many note onsets are not accompanied by ticks (**misses**; the fewer the better). For a given piece, the fit of a grid is a function of that grid's unit (inter-tick interval) and phase (where the ticks start relative to the music). The levels of a metric hierarchy in GTTM can be thought of as grids with the same phase and different units. Povel's insight provides a way, in principle, of quantifying the notion of the tactus, an aspect of GTTM theory that is of great psychological importance. We suggest that the tactus is that level (grid) in the metric hierarchy that maximizes the ratio of positive evidence (hits) to negative evidence (misses).

Each of the above models produces as output some aspect of the metric or rhythmic structure: LISTENER extracts rhythmic groups or phrases; the

programs by Longuet-Higgins, Lee, and Steedman extract the time signature, with some additional information about the grouping structure within each measure; Povel's program identifies a single metric unit as the beat or tactus. None of these simulations provides a complete analysis of the metric hierarchy. Furthermore, only Longuet-Higgins & Lee provide a psychologically plausible model of how listeners discover metric structure. The present work addresses these shortcomings.

Where Povel's program selects, out of a large set, a single best-fitting grid, our model generates a family of grids representing a metric hierarchy that satisfies the GTTM metric well-formedness rules. In addition, this must be done as the program "listens" to the score rather than all-at-once. To this end, we have adapted the production rules developed by Longuet-Higgins & Lee (1982). We have added to these rules by providing: a way to generate the entire hierarchy; criteria for excluding levels generated by the rules but not acceptable in the context of GTTM; and a means of generating those levels not generated by Longuet-Higgins & Lee's rules but required by GTTM.

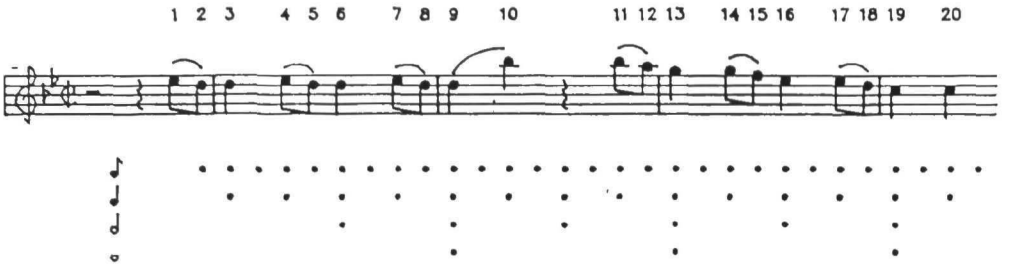
THE MODEL

In overview, our model has three types of processes: 1) Bottom-up processes that note time intervals between successive note onsets as they occur; 2) Top-down processes that take these intervals and combinations thereof and use them to predict the time of future events, with different intervals leading to different predictions. These predicted event times correspond to Povel's grids; and 3) Processes that evaluate the various predicted metric units or "grids" for consistency with a well-structured hierarchy as specified by GTTM.

Below is our analysis of Mozart's Symphony no. 40 (first violin part). It is comparable to GTTM's analysis (Lerdahl & Jackendoff, 1983, p. 23), with two differences. First, for reasons to be described, our analysis does not include the larger metric levels. Second, none of the metric levels in our analysis begins on the first note of the symphony, because doing so is not psychologically plausible. Once the program has identified metric levels, it generates expectations at each level, like a listener who has "got the beat." Getting the beat, however, takes some time. Once this is done, the program generates a metric structure that conforms to the well-formedness rules (WFRs): there is a dot at every note onset (WFR 1); if there is a dot at level L there is a dot at L-1 (WFR 2); dots at level L group dots at L-1 either by twos or by threes (WFR 3); temporal intervals (represented by distance) between dots at any level are uniform (WFR 4). Note also that strong beats (those with dots at higher levels) coincide with onsets of longer notes. This agrees with Mozart's bar lines and with preference rule 5.

The program begins by placing a marker, T1, at the onset of the first note. When it reaches the beginning of the second note, it places another marker, T2, and establishes the T2 - T1 interval as the current metric unit (MU). The hypothesis that this interval is a unit in the piece's metric structure predicts that another onset will occur one metric unit further on, so the program projects a third marker, T3, one MU into the future, i.e. at T2 + MU. The hypothesis is made explicit in a **metronome**, a process that is used to generate a level in the metric hierarchy. At this point the MU is an

eighth-note, so an eighth-note metronome is made. When this metronome "ticks," it places a dot in the growing metric hierarchy and then sets itself to tick again one MU later. The metronome counts its dots, and whether a dot coincides with a note onset (a hit).



Mozart's Symphony No. 40, First Movement (1st Violin part)

This completes processing at the onset of the second note. The next location is always the lesser of two distances: the distance to the next note onset and the distance to the next tick of the lowest level metronome. The distance to the sole metronome's next tick is the same as the distance to the next eighth-note, so the program moves to this location.

The program finds T3 at this new location, supporting the hypothesis that the eighth-note is a metric interval. On the strength of this, the program hypothesizes a higher-level temporal grouping at double the current (eighth-note) metric unit. There are two reasons for doubling rather than tripling the current MU to generate the next higher level. First, ratios of two are far more common. Second, two is the most common grouping in the subjective organization of identical, isochronous tones (Fraisse, 1982), and in spontaneous tapping (Fraisse, 1947-1948). The doubling operation (Longuet-Higgins & Lee's "*Conflate*" procedure) holds T1 fixed and moves T2 to where T3 is (i.e. at T1 + 2MU), recalculates the MU (T2 - T1) and projects T3 to T2 + MU. The new quarter-note MU generates a new metronome which is set to tick at T1 + MU, i.e. T2. There are now two metronomes, with different units. The program examines each in turn to see if it is due to tick at the current location. If it is, the metronome produces a dot, counts it, and scores a hit if there is a note onset at the current location. If a metronome is not due to tick but there is a note onset, the metronome records a miss, an onset it did not predict. In our example, both metronomes are due to tick at the current location (the third note onset), so now the metric hierarchy contains two dots at the eighth-note level and one at the quarter-note level, all hits.

The distance to the next note onset is a quarter-note, while the unit of the lowest metronome is an eighth-note, so the next location is halfway between the onset of note 3 and the onset of note 4. The program examines each metronome. Only the eighth-note metronome is set to tick here, and it produces a dot, but there is no onset here, so it is neither a hit nor a miss.

T3 occurs at the onset of the fourth note, so we might expect the program again to double the MU and create a new metronome. However, something more important has happened. At the onset of note 4, the program reaches the end of the note 3 quarter-note, and it recognizes that this note is longer than any it has heard before. Since longer notes usually initiate higher-level metric groupings (Povel & Essens, 1985), and since it is still early in the

piece, the program retrospectively interprets the first two notes as upbeats to the third note. Accordingly, an "*Update*" process moves T1 forward so as to make the onset of note 3 the anchor point of the metronomes, and projects T2 and T3 at one and two MUs, respectively, from T1. The MU is not changed, and no new metronome is created. Note that the *Conflation* that seemed warranted at this location is no longer possible, since T3 has been projected to a point we have not yet reached. The program now examines the metronomes: both are due to tick, and since there is an onset here both dots are hits.

At the onset of note 5 the eighth-note metronome yields a hit and the quarter-note a miss. At the onset of note 6, we have reached T3. The program therefore *Conflates*, doubling the quarter-note MU to a half-note, and makes a third metronome. This metronome is set to tick at T1 + MU, which happens to be where we are. Note that if T1 had not been Updated before this *Conflation*, the first dot at the half-note level would have been at the onset of note 4, which would have been inappropriate. Instead, the half-note metronome produces dots such that every other dot coincides with a bar line in the score. At note 9 the program reaches T3 once again, where another conflation yields a MU of a whole-note and a fourth metronome.

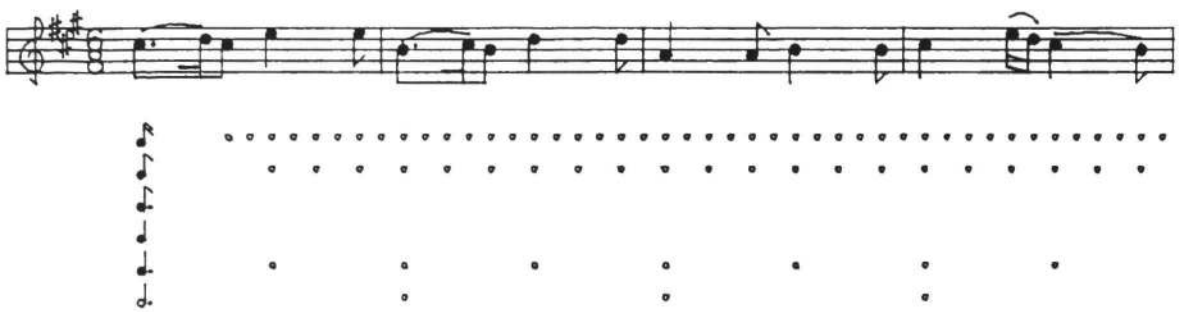
We do not allow enlargement of the MU beyond one whole-note, so our analysis does not produce GTTM's two-measure, four-measure and higher metric levels. The point of this limitation is that higher levels are: a) less perceptually salient (they are far above the tactus); and b) are better understood as defining phrasal boundaries than metric units (Longuet-Higgins & Lee, 1982). Indeed, GTTM allows for metric discontinuities at high levels and for violation of WFR 4 (which normally requires dots at a given level to be equally spaced in time) in such cases. We have not yet attempted to incorporate such intuitions in our program.

The rest of the analysis generates the full metric hierarchy and counts the hits and misses at each of its levels. In addition, a procedure, *Slide*, which we have added to those of Longuet-Higgins & Lee, slides all of the T-markers forward 1 MU whenever the program catches up with T3. We assume that the listener continues to predict future onsets on the basis of the largest experienced metric unit, and uses this projection to detect metric changes. When a metric change occurs, the framework of T-markers may be reset at the point of change. We have not yet implemented these intuitions, but the problem of meter change will be an important test of our model.

Our analysis of Mozart's Piano Sonata K.331, shown below, illustrates another important operation. When the program gets to note 2 it puts T2 there and declares the MU to be a dotted eighth-note. Before reaching T3 the program realizes that note 3 is longer than the note beginning at T2, i.e. note 2. Again, on the assumption that longer notes should initiate higher level metric units, the MU is now *Stretched* to a quarter-note by moving T2 to the onset of note 3. Intuitively, the purpose of the *Stretch* procedure is to handle dotted notes. A dotted note is usually followed by a complementary note which, added to the dotted note, yields a duration that fits the metric hierarchy at a higher level than either of the notes alone.

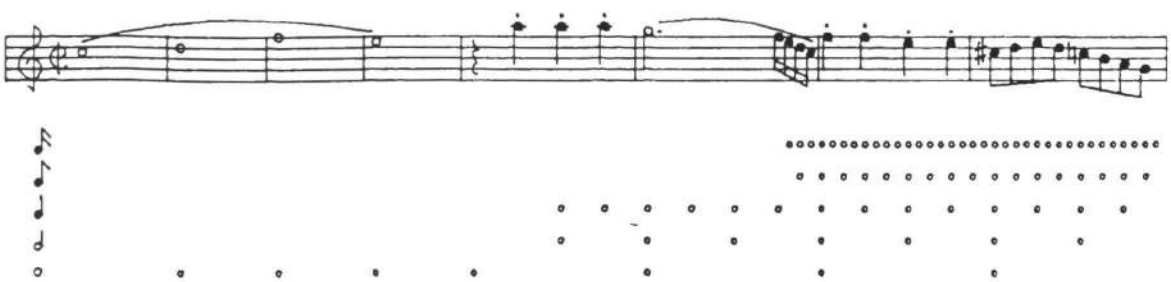
The program has so far created sixteenth-, dotted eighth- and quarter-note metronomes. By the end of note 4, it again finds that its hypothesis

(that there will be an onset at T3) disconfirmed, since note 4 surrounds T3. This suggests that the MU that generated that hypothesis is incorrect, and that the MU should be enlarged. The most conservative enlargement is made by moving T2 to the onset of note 4, yielding a MU of a dotted quarter-note. Both applications of the *Stretch* procedure illustrate a common principle: whenever the metric unit is enlarged by some means other than doubling or tripling it is assumed that the old MU is incorrect and should therefore be eliminated (by marking it as rejected). This indicates that it is not part of the listener's metric representation.



Mozart's Piano Sonata, K. 331, First Movement

In some pieces the listener may hear a note whose duration is not represented in the metric hierarchy. In such cases we assume that the listener creates a metronome to represent that duration if it is consistent with the current MU. By consistent we mean that the duration in question is an integral multiple or divisor of the MU. Consider the beginning of the last movement of Mozart's Symphony no. 41 ("Jupiter") below. At the outset the listener establishes a MU of a whole-note, and eventually hears the first of the quarter-notes. Longuet-Higgins & Lee's rules do not provide any way to generate smaller metric levels, but we assume that the music itself may directly dictate levels in such cases. Since the first quarter-note is consistent with the MU, the program creates a quarter-note metronome. A few notes later we hear a dotted half-note; this, too, is smaller than the MU, but it is not consistent, so no metronome is made.



Mozart's Symphony No. 41, Last Movement

The "Jupiter" example also illustrates a third way that metric levels are created. For the first four bars the metric hierarchy consists of a single level. When the quarter-note metronome is created, its unit is consistent

with the current MU, but the metric hierarchy now violates WFR 3, which limits the ratio between adjacent levels to three or less. The solution is to interpolate a third level that satisfies WFR 3 with respect to both the quarter-note level and the whole-note level, i.e. a half-note level. After the dotted half, we hear a sixteenth-note, and it is again necessary to create a new metronome, interpolated at the eighth-note level.

CONCLUSION

Our program yields a psychologically and musically plausible metric analysis of a wide variety of scores. Using note-by-note processing, it produces a metric hierarchy that conforms to the GTTM rules, and it identifies one of the metric levels as the **tactus** or beat. There are many other scores, however, that it cannot correctly analyze. This is a limitation of duration-only analysis: rhythm is not the sole carrier of information about metric structure. The extent to which our approach succeeds reflects the redundancy between rhythmic, melodic and harmonic dimensions in most music. The important question, therefore, is whether those scores that our program cannot analyze are also difficult for human listeners when only the rhythm is presented. Answering this question will be an important test of our model.

ACKNOWLEDGMENTS

This work was supported by an NSF Pre-Doctoral Fellowship to Miller and a PSC-CUNY Faculty Research Award to Jones.

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.

Longuet-Higgins, H.C. & Steedman, M.J. (1971). On interpreting Bach. In D. Michie and B. Meltzer (Eds.), *Machine Intelligence 6.* Edinburgh: Edinburgh University Press.

Povel, D.-J. (1984). A theoretical framework for rhythm perception. *Psychological Research, 45,* 315-337.

Povel, D.-J. & Essens, P. (1985). Perception of temporal patterns. *Music Perception, 2,* 411-440.

Simon, H.A. (1968). Perception du pattern musical par AUDITEUR. *Sciences de l'Art, V-2,* 28-34.

Steedman, M.J. (1977). The perception of musical rhythm and metre. *Perception, 6,* 555-569.

Acquiring Computer Skills by Exploration versus Demonstration

Franz Schmalhofer and Otto Kühn[1]

McGill University / Canada, and Universität Freiburg / West Germany

It is well known that more effective learning can be achieved by tailoring the learning episodes to the particular needs of an individual rather than presenting the same sequence of instructions to all learners. There are two ways by which this can be achieved: a tutor can adjust its instructions to the learner's previously acquired knowledge or it may simply allow the learners themselves to determine the sequence of learning episodes. In the first case, the individualization is determined by the prior learning histories with the global learning goals being assumed identical for all learners. The LISP tutor of Anderson, Boyle & Reiser (1984) is an example for such a system, in which the learning goals are determined by the tutoring system, and the instructions which assist the learners in performing the respective task are adjusted to the learners' knowledge. Further individualization can be achieved by having the learners learn by exploration: the learners themselves can now set their own learning goals according to their specific interests. Student-driven exploration can be enhanced by providing instructions on those occasions where learning by exploration fails. For example, the redundancy checkers discussed by Brown (1984), can be used to detect weaknesses in a student's exploration.

In a number of cases learning by exploration may be more appropriate. For example, in application systems such as text editors or spreadsheets depending on their needs users may learn different parts of the system. Under such circumstances it can be quite frustrating to the learners when they are taught system components in which they are not interested. Contrary to instruction-based learning, in which advanced learners may still be presented with introductory materials due to the difficulties of knowledge diagnosis, learning by exploration allows the learners themselves to decide what to learn. Under certain circumstances learning by exploration may well be more effective then learning from instruction. For example Carroll et al. (1985) have shown that a text editing system may be learned more effectively by exploring it than by studying a conventional manual.

The advantages of learning by exploration may be caused by a number of different factors. Learners can selectively acquire that knowledge which they consider most important. They can be more active and set their own learning goals. In order to achieve their learning goals, they can engage in problem solving (Robert, 1986). This may lead to procedural and problem solving oriented knowledge representations which are better suited for solving computer tasks. Successfully solving these problems may be quite motivating for the learner. Since learning by exploration originates from the student's own domain knowledge, the newly acquired knowledge becomes inherently connected and interwoven with the prior knowledge. Therefore, it may be better remembered.

However, each of these advantages may also turn into a disadvantage. A student could have insufficient domain knowledge to set appropriate learning goals. Because of insufficient domain knowledge the students may not be able to determine which knowledge is really important. They may acquire suboptimal procedures for achieving their goals or in the extreme case no successful procedures at all. Problem solving processes may not always be successfully completed and can be more time consuming than learning from instructions. This causes frustration for the learner. A student's lack of domain knowledge can thus put severe limitations on what can be learned by exploration.

Although learning by exploration and learning by instructions (or, more specifically, demonstrations) differ in a number of interrelated ways, one difference appears to be most fundamental. While instructional materials are determined by the teacher, who is very knowledgeable of the domain, in exploration the learning episodes are generated by the students who know about their particular knowledge desires. The advantages and disadvantages of learner versus teacher generated learning episodes were investigated in an experiment in which 80 students from the University of Freiburg acquired some elementary LISP programming knowledge. The results of this study show how the effectiveness of

[1] This research was supported by grant Schm 648/1 from DFG (German Science Foundation)

learning by exploration depends upon the amount of relevant knowledge which the learner can utilize for generating learning episodes. These results suggest a particular combination of the two learning methods. On this basis, a supervised exploration environment for acquiring some elementary LISP knowledge was developed, implemented on an IBM PC/AT, and consequently evaluated through two learners' think aloud protocols.

Experiment

Forty students who had some previous experience with computers (either they had taken a BASIC programming course or had used some application software such as word processors or database programs) and 40 students without any prior computer experience were first instructed about some fundamental LISP concepts (atoms and lists). Then they acquired additional knowledge about some simple LISP functions either by exploration or by learning from demonstration examples. In particular the function FIRST, which extracts the first element from a list, and the function SET, which binds a LISP-expression to some symbol, were learned. Simple LISP functions were used as the learning domain because modularity is a prerequesite of explorability and the LISP functions satisfy this requirement. Also, the subjects of the study should not have had any specific domain knowledge. Whereas this was true for LISP, they may have already been familiar with text processing.

Exploration condition: In the exploration condition the learners could enter LISP expressions with an editor providing help for generating syntactically correct inputs. A LISP interpreter evaluated these expressions and gave appropriate feedback. The exploration condition was divided into three blocks so that each of the functions which were to be learned would be noticed by the subjects. For each block the subjects generated either 8 inputs (first and second block) or 16 inputs (third block). At the beginning of each block one or two simple meaningful inputs to the LISP system were presented, namely

1. "(FIRST '(A B))",
2. "(FIRST (FIRST '((A B) C)))",
3. "(SET 'FRIENDS '(JACK JOHN)) (FIRST FRIENDS)".

Instruction condition: In the instruction condition 32 appropriately selected examples were presented where each block started with the example which was also presented in the exploration condition. In each condition 32 training examples were thus generated or presented. The exploration subjects who entered the presented inputs had to create another 28 inputs on their own, whereas the subjects in the demonstration condition were presented with 32 examples and could not generate any examples by themselves.

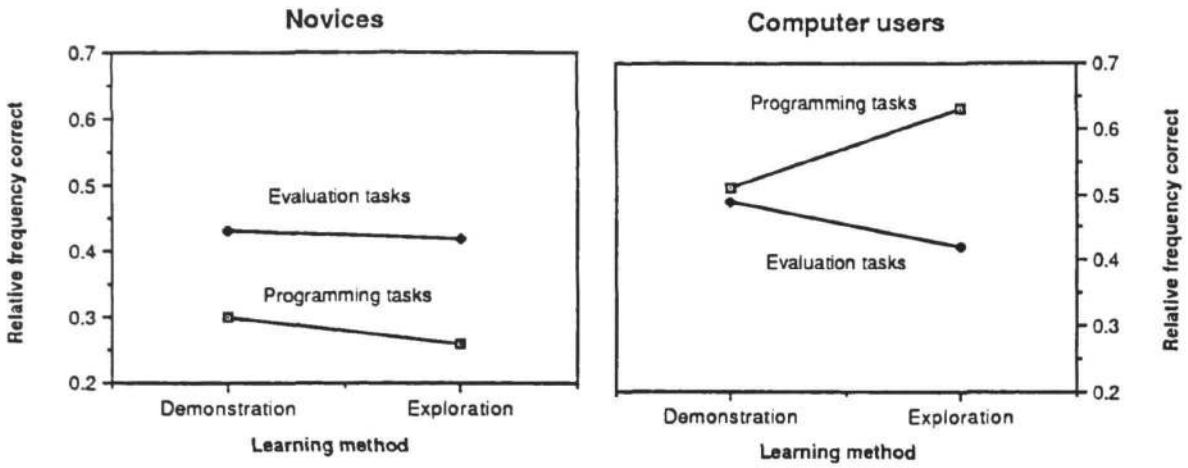
Programming and evaluation tasks: The acquired knowledge of each learner was tested by 10 programming tasks in which the subjects had to generate an input to the LISP-system in order to obtain some prespecified result. The inputs were evaluated by the LISP-interpreter and the result was shown to the subjects. If the result of the subjects' input was not the result that was to be achieved, the subjects were given two more trials to achieve the correct result. Thereafter, the subjects' knowledge about the LISP-system was examined by evaluation tasks, in which inputs to the LISP-system were presented, and the subjects, rather than the LISP interpreter, had to generate the results. The whole experiment took between 1.25 and 3 hours. For a more detailed description see Kühn & Schmalhofer, 1987.

Results :

For the novices and the computer users the relative frequencies of correct solutions in the programming and the evaluation tasks as a function of instruction method are shown in Figure 1. A (2x2x2) ANOVA with the factors prior knowledge, instruction method and test task showed that overall the two tasks were about equally difficult ($F(1,76)=0.31$), and that computer users performed better ($F(1,76)=21.6$, $MSE=0.35$, $p<.001$). More interestingly, novices performed better in the evaluation tasks and computer users performed better in the programming tasks, resulting in a prior knowledge by test task interaction ($F(1,76)=20.5$, $MSE=0.14$, $p<.001$). In addition, learning from demonstrations was more useful for correctly solving the evaluation tasks and learning by exploration was more effective for the programming tasks, resulting in an instruction method by test task interaction ($F(1,76)=5.13$, $MSE=0.14$, $p<.05$). Supposedly, in the exploration condition the generation of inputs was trained which is an important component for successfully solving programming tasks. The instruction groups had some

advantage in the evaluation tasks, possibly because the self generated learning examples provide less complete information about the system than the examples selected by a teacher.

Figure 1: Proportions of correctly solved test tasks



In order to analyse the relation between the subjects (programming and evaluation) performance and the studied examples, the training examples generated by the subjects in the exploration condition as well as the training examples presented by the tutor in the instruction condition were classified as belonging to one of four categories which were definied as follows: 1) positive examples that contain new information about the system, 2) redundant positive examples, 3) "near misses" (Winston, 1987), i.e. negative examples that are very similar to positive examples and thus convey information about the system, and 4) all other inputs were classified as "far misses". This classification was performed with a LISP programme, which constructed generalized templates representing the knowledge that may be acquired from the training examples (see Schmalhofer, 1986). The classification was performed according to the model of knowledge acquisition of the learning environment that is presented in the next section of this paper.

Table 1 shows the relative frequencies of the four types of examples generated by the novices, the computer users and the tutor. It can be seen that the exploration subjects generated fewer negative examples than were presented in the instruction condition. Both novices and computer users generated more redundant inputs than were presented in the instruction condition. Also, computer users generated more positive new examples than novices ($t(38)=2.45$, $p<.05$). Furthermore, a considerable proportion of the generated inputs in the exploration condition were far misses which do not provide useful information.

Although the training examples generated in the exploration condition were of poorer quality than those presented in the instruction condition, the computer users of the exploration condition performed better in the programming tasks than the computer users of the instruction condition. For the programming tasks, the advantage of generating training examples apparently can outweigh the disadvantage due to the usually poorer quality of selfgenerated training examples.

Table 1: Proportions of 4 types of training examples as generated by novices and computer users, or presented in the instruction condition

Type of example :	*Novices*	*Computer Users*	*Instruction*
positive new	*0.23*	*0.31*	*0.38*
positive redundant	*0.28*	*0.31*	*0.13*
"near misses"	*0.24*	*0.19*	*0.50*
"far misses"	*0.26*	*0.19*	*0.00*

Thus, the knowledge that can be acquired from learning by exploration depends upon the quality of the generated training examples, which itself depends upon the subjects' prior knowledge. Two multiple regression analyses were conducted for the proportions of correctly solved programming and evaluation tasks with the proportions of the first 3 types of training examples (positve new, positive redundant, and near misses) and prior knowledge (with the dummy coding 0 for novices and 1 for computer users) as predictors. Initially, all 4 predictors were entered into the regression equation and then insignificant predictors were dropped stepwise. The results of these analysis are shown in Table 2.

Table 2: Prediction of correctly solved tasks from prior knowledge and 3 types of generated training examples in the exploration condition

	programming tasks		evaluation tasks	
	ß-weight	correlation	ß-weight	correlation
prior knowledge	0.290***	0.605***	0.104*	0.457**
positive new	1.328***	0.621***	0.399*	0.433*
positive redundant	-0.421+	-0.038		

*** = p<.001, ** = p<.01, * = p<.05, + = p<.10

It can be seen that the proportion of positive new training examples is a good predictor of the performance in the two tasks, even after the effect due to differences in prior knowledge has been taken into account. The results furthermore show that the more redundant training examples were generated the less programming tasks could be solved in the test phase. The results demonstrate that the effectiveness of learning by exploration depends on the learners' domain knowledge and their ability to generate appropriate training examples .

Combining exploration and instruction in a tutoring component

The experimental results indicate that learners with sufficient prior knowledge benefit from exploring a computer system, and that instructions are more efficient when that knowledge is not yet available. Thus instructions can be used to induce the knowledge which is necessary for more successful exploration. The correlation between the type of generated training example and task performance shows that the generated training examples can be used to diagnose whether the learner has sufficient prior knowledge for successful exploration. On the basis of such a diagnosis specific instructions can be presented to provide the information which is needed to make exploration successful.

Rather than starting learning by instruction it thus seems feasible to have all learners start by exploring the system, monitor their exploration behavior and provide instructions as needed. Thereby the learners can engage in active learning and determine themselves what to learn. Instructions are only presented when they are needed to maintain this active learning process. The advantages of learning by exploration can thus be utilized while its disadvantages are avoided. Such an exploration environment was developed for learning some elementary LISP functions.

The basic exploration environment:

The learning environment is based upon a reduced LISP interpreter which is written in TURBO PASCAL on an IBM PC/AT. It can handle the functions LIST, FIRST, REST, SET, DEFUN as well as any combination and any list structure. By acting in the learning environment the student should learn:

- the number and type of arguments which a function requires
- the correct syntax for an input to the LISP system
- how a given input is evaluated and what result is returned
- that quoted expressions, bound atoms or function calls can be specified as arguments.

At the beginning of the learning session only the names of the functions which the learner can explore are shown on the top of the screen. The learner must then generate an input to the LISP system. In order to avoid unnecessary typing errors, only characters that are valid in LISP (letters, digits, blank, parentheses and the quote) can be typed, and only lines with balanced parentheses are accepted as inputs. In addition colors are used to indicate the level of nesting in the expressions. As these features help generating syntactically correct training examples, they should reduce the number of trivial syntax errors

that result from typing errors and limitations of a learner's memory. Such errors don't convey useful information about the system to be studied, and they reduce the efficiency of learning by exploration by increasing the study time (see Köhne & Weber, 1987; Waloszek, Weber, Wender, 1986)

The generated inputs are evaluated by the LISP interpreter and the result of the evaluation or an error message are displayed. If the monitor which supervises the learning process detects a sequence of training examples that supposedly do not contain useful information, the learner is prompted to press a key in order to get help. The detection of inefficient exploration as well as the assistance that is provided are based upon the monitoring of the knowledge acquisition process.

Monitoring of the knowledge acquisition process

Every input of a student is immediately analyzed by the monitor. Ideally such an analysis should be conducted according to psychological principles. In particular it should render a description which is closely related to the information which the learners store in their memories. It is known that rather than individually storing every example in memory learners only remember those things which they consider generally relevant and ignore the very specific information. The knowledge of the general form of correct inputs to the LISP system can be described by templates (Anderson, Farrell & Sauers, 1984).

The monitor models a template construction process by an inductive learning mechanism which creates increasingly general template representations. For the first and all other inputs the LISP interpreter determines whether or not it is syntactically correct. Syntactically correct inputs are called positive examples. The first positive example is stored in memory. When a second positive example is generated the two examples are compared from left to right in order to construct a template. As long as the respective elements of the two examples are equal they are taken as constants of the template. When they are unequal but are named as belonging to the same class, a variable is introduced into the template with the constraint that it may take as a value any member of the respective class. If the two elements that are being compared belong to different classes it is checked whether or not they both belong to a more general superclass. If this is the case, a variable is introduced with the constraint that it must be bound to a member of the respective superclass. If no common superclass can be found, the generated input is used to build a separate template.

Since the generated inputs may differ in the number of elements, generalizations are made not only with respect to the class membership, but also with respect to the number of elements. The latter generalizations are performed as follows: If during the comparison from left to right either in the input or the template all elements have been processed, while in the other additional elements are available, the class memberships of these elements are determined, and if a common supertype can be found, an additional variable is added to the generalized template which can match any number of elements of that class; if no common class can be determined, different variables are used. For a sequence of input examples Table 3 shows the template, which is constructed from the first example and how this template is modified and a separate template is constructed from the forth example. (?A denotes a single member and +A an arbitrary number of members of a class.)

Table 3: Templates formed from a sequence of inputs

input sequence	Constructed templates and modifications
1. (FIRST '(A B))	(FIRST '(A 'B)), (A is-atom), (B is-atom)
2. (FIRST '(X (Y Z))	(FIRST '(?A '?B)), (?A is-atom), (?B is-expr)
3. (FIRST '((A B)))	(FIRST '(?A +B)), (?A is-expr), (+B is-expr)
4. (FIRST FRIENDS)	(FIRST ?A), (?A is-bound-atom)
	(FIRST '(?A +B)), (?A is-expr), (+B is-expr)

A think-aloud study showed, that three causes of negative examples can be distinguished: 1) accidental errors such as misspelling a function name or forgetting to type a quote, 2) errors that are made to determine what elements of a template are necessary and whether it can be further generalized, and 3) errors that occur when a learner attempts new and more complex inputs.The verbal protocols showed that for incorrect inputs of type 1) or 2), the error message provided by the system contained sufficient information for the learner. Therefore, no intervention of the tutor is required. It can therefore be assumed that a negative input of type 1) or 2) is likely to be followed by a positive input. When learners make errors while trying something new, the system error message may not provide enough information. In this case,

a learner will produce a sequence of errors. Since only type 3 errors are likely to occurr in a sequence without intermittent positive examples, they can be easily detected by the monitor.

Providing examples and text-information to assist exploration

Assistance to learning by exploration is provided by the tutor on two occasions: 1) when a sequence of n (=2 in the current version of the tutor) redundant inputs has been detected, or 2) when a sequence of m (=2) errors has been detected.

Since redundant examples are usually generated when a learner does not know what else can be learned about the system, information about more general or presently not yet explored features of the system should be provided. Such information can best be provided in the form of a short text. The template that matched the last example is examined as to whether it can be further generalized or whether it already describes some unit of the expert knowledge. If further generalization is possible, a verbal description of more general inputs is presented. If no further generalization of the particular template is possible, a general verbal description of the not yet learned functions is presented. The verbal descriptions that are provided as help are all prestored so that they can simply be selected for presentation. Table 4 shows the help information for some redundant examples.

Table 4: Help information for some redundant inputs

example 1: *(FIRST '(A B))*
example 2: *(FIRST '(X Y))*
help information: The argument for the function FIRST can be a list of any complexity.
example 1: *(FIRST '((A B) (C D)))*
example 2: *(FIRST '(X Y))*
help information: The argument for the function FIRST can also be a bound atom or a function call.

When a sequence of errors is detected, it can be assumed that the learner wanted to perform a more complex task and did not know how to correctly specify the parameters of the task. Based upon the PUPS theory (Anderson & Thompson, 1987), it may be suspected that such errors occur because the specific form is unknown. A form can best be taught by giving an example. Since the learner wanted to generate a particular example the correct form for that particular example should be presented. To accomplish this goal, the last incorrect input of the error sequence is corrected and then presented to the learner. The correction is accomplished by analyzing the incorrect input from left to right. Parenthese and quotes are deleted or added if needed, with the following restriction: If a symbol is identified as being a function name, the input is corrected whenever possible in a way so that the function name yields a function call. Table 5 shows how some incorrect inputs are corrected.

Table 5: Correction of negative examples

incorrect:	corrected:
(FIRST (FIRST '(a (b))))	(FIRST (FIRST '((a) (b))))
(FIRST (REST '(a b)) (REST c d))	(FIRST (REST '(a b (rest c d))))
(LIST (FIRST '(a b) (REST '(c d))))	(LIST (FIRST '(a b)) (REST '(c d)))
I AM HERE	'(I AM HERE)

A preliminary evaluation of the appropriateness of the provided instructions was performed by having two subjects, who were first instructed about data representations in LISP, think aloud while learning the elementary LISP functions by exploration. They were instructed to explore as long as they found it to be a useful learning experience, which was about 1.5 hours. In order not to interfere with the learners' usual exploration, no tutor assistance was provided to the learners. The sequences of the subjects interactions with the LISP-system were recorded and then used to determine the first learning episodes of each block where assistance would have been provided by the tutor. Only the first occasions of each of the 16 blocks in which the tutor would have provided assistance were used for evaluating the appropriateness of the tutor's assistance. In one block the tutor would not have provided assistance, so that only 15 assistances were to be evaluated. In 10 cases explanatory text and in 5 cases corrected examples would

have been presented. The tutor's assistance was then compared to the learners' verbalisations, which were used to judge whether the tutor assistance would have been adequate. The tutor's assistance was judged as helpful in 6 cases, as neutral in 6 cases, and as inappropriate in 3 cases. In two of the 6 helpful cases the tutor's corrected example would have prevented an error path of nine episodes, which may be judged to be quite effective.

Discussion

Most Intelligent Tutoring Systems are more instruction- than exploration-based and provide the student with rather little possibility to learn by exploration. However, for acquiring computer knowledge learning by exploration may be quite fruitful. The studied material may be better remembered when learning by exploration because students themselves can select what to learn and because the learning material originates from the students' own memory, and for whatever reason self-generated information is better remembered (Slamecka & Katsaiti, 1987). With respect to performing simple programming tasks this prediction was confirmed for learners who had some very general prior domain knowledge (computer users), but not for complete computer-novices. Learning by exploration allows to practise the generation of possible system inputs, which appears to be an important computer skill component. We suggest that learning by exploration can become even more effective when a learner's exploration is monitored so that often occuring problems can be detected and bypassed by giving appropriate instructions. A supervised exploration environment for the learning of some LISP-basics was described. In this environment explanatory text and an error-correction facility are used to provied learners with general information as well as with information about the specific forms of system-inputs. A preliminary empirical evaluation was performed for an implementation on a PC.

We believe that supervised exploration may be particularly suited for learning the specific forms for coding some already known procedures in a new environment. This may occur when writing a program in a new programming language or using some new application software. Since the commands of a system may be separately learned, they can be learned by exploration and may be even better learned by supervised exploration. However, instruction-based learning may be necessary for acquiring more complex knowledge. Learning by supervised exploration is therefore only proposed as a component for Intelligent Tutoring Systems.

References

Anderson, J.R. Boyle, C.F. & Reiser, B. (1985). Intelligent Tutoring Systems. Science, 228, 456-462.

Anderson, J.R., Farrell, R., & Sauers, R. (1986). Learning to program in LISP. Cognititve Science, 1984, 8, 87-129.

Anderson, J.R. & Thompson, R. (1987). Knowledge Representation in the PUPS Theory. Technical Report, Department of Psychology, Carnegie-Mellon University.

Brown, J.S. (1984). Learning -by-doing revisited for electronic environments. In White, M.A. Ed, The Future of Electronic Learning. New Jersey: Lawrence Erlbaum Associates.

Carroll, J.M., Mack, R.L. & Lewis, C.H. (1985) Exploring exploring a word processor. Human-Computer-Interaction, 1985, 1, 283-307.

Köhne, A. & Weber, G. (1987). Struedi: A LISP-Structure Editor for Novice Programmers. In: Bullinger, H.J. & Shackel, B. (ed.). Human-Computer Interaction INTERACT'87. Elsevier Science Publishers B.V. (North-Holland).

Kühn, O. & Schmalhofer, F. (1987). Erlernen der Computerbenutzung: durch gezielt sequenzierte Instruktion oder durch Explorieren? In: Schönpflug, W. (ed). Software Ergonomie 87, Teubner Verlag, Stuttgart, 387-397.

Slamecka, N.J. & Katsaiti, L.T. (1987). The Generation Effect as an Artifact of Selective Displaced Rehearsal. Journal of Memory and Language, 26, 589-607.

Waloszek, G., Weber, G. & Wender, K.F. (1986). Entwicklung eines intelligenten LISP-Tutors (Bericht No. 2). Braunschweig: Technische Universität, Institut für Psychologie.

Winston, P.H. (1984) Artificial Intelligence. Second Edition. Addison-Wesley, Reading, Massachsetts.

A HYBRID MODEL FOR CONTROLLING RETRIEVAL OF EPISODES

Colleen M. Seifert

UCSD and NPRDC

A central problem for models of memory is the retrieval of episodes. Most models assume retrieval is an automatic process where an input is matched to the contents of memory, and an episode is activated based on similarity. If there is a high degree of similarity between an input and a particular case, and both cases share little with other cases in memory, then similarity alone may be enough to account for the spontaneous retrieval of the case. However, when there are many instances that overlap in similarity, or when the similarities are abstract in nature, it appears that predicting retrieval based on similarity measures alone is quite difficult. Schank (1982) argues that the process of reminding is *mediated* by an abstract knowledge structure used to understand the original event -- cases are activated as a consequence of activating organizing schemas that capture important similarities. Is similarity alone enough to cause the automatic activation of prior episodes? In the next sections, I will present evidence from empirical studies about the conditions under which such remindings occur; specifically, that reminding based on similarity is not an automatic process but involves a strategic or goal-based function. The same information, in the presence of different processing goals such as explanation or planning, will result in retrieval of different prior cases from memory. In order to account for this, I propose a hybrid model of retrieval that incorporates both the content-addressable character of distributed memory models with the controlling influence of goals in processing. In the model, goals act as a controlling mechanism to focus attention on features that will result in retrieval of relevant episodes.

Experimental Evidence for Activation of Episodes

The question of interest is whether memory organization (similarity) alone will predict the automatic activation of cases. Leaving aside the question of conscious awareness of the result of the retrieval, we can investigate whether the memory organization formed at encoding provides connections between cases such that activating one case in memory will result in the activation of other cases encoded with the same organizing structures. This activation may be the result of an automatic retrieval process, that is uncontrolled and non-optional, as in semantic priming, or it may occur only under some conditions, under strategic control.

The basic paradigm involved presenting pairs of stories that either share or do not share the organizing knowledge structure, and then testing whether the similarity-based memory organization that results causes different activation patterns. In the studies, time to answer a question about a previously read story is used to measure the accessibility of the story in memory. If the two stories are connected by the organizing structure in memory, then answering a question from one story should activate the other story in memory, resulting in faster responses to a question from that story.

Experiments with abstract thematic structures

The materials chosen for the experiments (Seifert, McKoon, Abelson, and Ratcliff, 1985), included similarities based on the types of structures evident in protocols of natural remindings; namely, the goal and plan interaction that occur as goals are pursued. For example, the adage "closing the barn door before the

horse is gone" can be characterized as a planning failure where X knows a plan to prevent goal failure, but delays execution to avoid the cost until the goal is failing; then, the plan is executed, but fails because it is not a recovery plan but a prevention plan (Dyer, 1983). If two stories based on the same theme are connected in memory, responding to an item from one should speed the time to respond to an item from the other, compared to where the preceding item refers to a story that does not share the same theme, and therefore should have no connection in memory between the two stories. The results showed that the shared knowledge structure did not affect the ease of access of the episodes in memory. However, in a second experiment we instructed the subject to think about the theme as they read, and to rate the similarity of the story pair after the test list. This time, there was a significant effect of thematic similarity on response times, in that responses were faster for test items when the story pair shared the same thematic organizing structure. The only difference between the two experiments was the instruction to attend to similarity. We conclude, then, that accessing the same schema does not automatically connect two episodes in memory; instead, some strategic purpose is necessary to promote activation of prior episodes. The strategic basis for the ability to utilize connections between episodes has been replicated in other experiments (Seifert, McKoon, Abelson, and Ratcliff, 1985).

Experiments with content-based structures

One question was whether the strategic result was specific to abstract structures, or would hold true of more content based structures. To test this, we examined the effect of memory organization packets (MOPs) (Schank, 1982) on connections between cases (for a complete description, see McKoon, Ratcliff, and Seifert, 1988). Pairs of stories were written to instantiate the same MOP (such as "going to the beach") but without overlapping lexical items in the description of MOP actions. In the experimental procedure, subjects read a long list of stories, and then, after reading all the stories, connections from one story to another were measured by priming in old/new recognition judgments of phrases from the stories. Since the test items included MOP information, any activation effects may be due to connections in semantic memory, and not reflect any activation from one case to another case. In a second experiment, the test items used as primes contained only information that refers to the story-specific representation. Results showed priming from one story to another story of the same MOP when test phrases were related to the MOP; however, when the test phrases were story-specific but not related to the MOP, there was no evidence of priming. One way to describe the results of these experiments is to say that subjects cannot discriminate whether two MOP-related phrases were from the same or different stories; for example, "spreading out her towel in a dry place" from one beach story and "found an empty space for her blanket" from another beach story are equally good primes for "slowly strolled into the cool ocean". The results as a whole demonstrate that case to case activation is not an automatic phenomenon resulting from connections in memory.

A paradigm for intentional reminding

Since a strategic process appears to be involved in the activation of cases, a next step was to look at reminding within a strategy-based task. Up to this point, in order to keep the subject unaware of the intent of the experiment, we examined activation rather than conscious reminding. Now that the activation appears to be dependent upon a strategic process, we can examine reminding in a paradigm more similar to natural remindings. In these experiments (Seifert, McKoon, Abelson, and Ratcliff, 1985), subjects were asked to study a set of stories, then to read a new set of stories followed by test items from the study set. The test stories had either the same theme (as described in the earlier experiments) or a

different theme than the story that the target item was from. The results showed a significant facilitation effect for same theme pairs. A test story appeared to activate a previous story based on its thematic similarity, resulting in response facilitation. Subjects reported that as they read the test stories, studied stories occasionally "came to mind"; if they did, they were a good predictor of the target item, and so the reminding was useful information to the task. It seems the strategic aspect of this task was that the purpose of the reminding was built into it -- usefulness in predicting the test item provided a functional purpose for the reminding. Also, subjects were conscious of the remindings and their utility, and this may have encouraged attention to themes and the resulting remindings. This methodology was the first to investigate reminding in the laboratory.

Incorporating Goals in Reminding

From the experiments presented above, I conclude that encoding a story does not always activate a thematically similar story *unless* there is also present a functional or strategic purpose for the reminding. These results are reminiscent of analogical transfer studies (Gick and Holyoak, 1983), where transfer of a story solution to a new problem was infrequent unless either instructions or multiple example stories were provided. The similarities in the experiments reported here are more obvious than those in the transfer literature, and yet subjects were not reminded unless encouraged by the strategic nature of the task. Intention appears to play a bigger role in retrieval than may have been assumed, because without it the retrieval does not occur. This strategic aspect of retrieval points out the importance of cognitive processing goals; apparently, such goals may act as a constraint on retrieval in that activation occurs that would not under other circumstances. In this sense, cognitive goals appear to form a context within which retrieval operates, and the nature of this context determines what remindings occur. The results suggest that, at the least, *intentional reminding* (Schank, 1982) plays a much bigger role than previously indicated. I define intentional very broadly, as not always consciously intended, but as a bias in processing.

A result that supports this conclusion is that the low transfer rates from a story scenario to a new problem (Gick and Holyoak, 1983) were increased when the context was made congruent in the two cases (K. Holyoak and L. Novik, personal communication, February 1988). That is, presenting the first case *as a problem* rather than as a "story understanding task" improved the spontaneous application of the prior solution to the new problem. Since processing goals appear to focus attention within the retrieval process, the types of goals and their connections to features must be examined. Situations that appear to foster remindings in humans are functions like explanation, planning, argumentation, decision making, and conversation. The kinds of processing goals that may be involved can be described at a general level, but certainly we can discover more about what subtasks of cognition may be involved (Chandrasekharan, 1987). For example, the understander seeks, in his processing of new input in conversation, to be reminded of a memory that relates to what he heard and provides evidence for the point of view he wishes to defend. Remindings can serve to verify your analysis of episode, illustrate why your reasoning is valid, justify or support a claim, give specific solution information, and perhaps provide an analogy.

A Distributed Model of Retrieval

I have argued from experimental results that strategic purpose is important because automatic processes like retrieval are mediated by processing goals. How might goals be incorporated into the reminding process? Retrieval can be modelled as a feature space with encoded episodes, as in a completion network in a parallel distributed processing model of memory (McClelland and Rumelhart, 1985). Memory access is determined by the similarity between input features and stored

patterns, where the process finds the activation pattern that bests fits the connection constraints. For example, given an input such as "a Professor forgot his computer password", a similarity match may find a particular episode pattern, such as "Prof. Jones forgot his password last week", to be a close enough match compared to other patterns in memory. If the result of the matching process is a set of features corresponding to a schema, the retrieved pattern might be "Absent minded Professors forget things." Another possibility is that nothing will match well enough for a stable pattern to be retrieved. These results can be characterized in terms of maximizing goodness of fit: a good fit may be an episode reminding, an adequate fit may a be a generalization, and a low fit may mean no close matches could be retrieved.

Rumelhart (1988) has proposed that the matching process can operate by relaxing the constraints on the match, resulting in analogy in cases where no distinct pattern could be retrieved by strict similarity to the input pattern. For example, if too many episodes share the feature set, or if some of the features are inconsistent with the patterns already encoded into the memory, then no episode pattern could be isolated by similarity alone. However, by softening the constraints on the match, some of the input features could be utilized as "don't care" constraints, allowing analogical matches that focus on some features and throw out features that may be preventing a match from occurring. The notion is to release features by allowing the network to "turn them off" progressively until a better match is found. Rumelhart's proposal is to feed additional input strength to the features differentially, and allow the system to find the overall best fit by overriding features with low values that conflict. A feature is thrown out only if doing so leads to better fit than the contribution from that feature adds to the match. This *relaxed match* process provides a mechanism for overriding features that are preventing a match (such as context information) while giving increased importance to features that may be helpful in analogical transfer to other episodes.

Methods of feature selection

The softening of constraints could be accomplished in several ways. The particular implementation Rumelhart proposes is to deliver a constant amount of activation to each of the features; the lower the constant, the more likely that feature's contribution to fit will be less useful than throwing it out, making it easier for the system to override it. The selection of *which* features to assign what values to is central to the solution of the relaxed match. One possibility is to select the values assigned to each feature at random, resulting in a blind focus to the match process. This is a feasible method, but we would expect then to generate a lot of episode remindings with nonsensical similarities and ignore important ones. For example, from the Professor example, we might retrive a reminding like "Joe owns a computer." The random selection problem is more impressive with more input features in the match; clearly, it would be easy to recall episodes that share features but without a coherent similarity -- a correlational match but not a causal one. It may be that human memory operates this way, but it most often appears that remindings result from coherent similarities.

Another method of assigning values to the input features is to order all the features along an abstraction hierarchy (Rumelhart, 1988). By weakening constraints on concrete features first, and then progressing to abstract features until reaching a match, greater weight is given to abstract features and more abstract analogies will result from the relaxed match process. This approach certainly seems plausible given the results from example remindings; however, there are several problems with an intrinsic ordering on feature importance. First, "abstract" is hard to define, as even thematic patterns like "closing the barn door after the horse has gone" can have pretty contentful features (i.e., "too late").

Though generality can be defined in terms of how many instances are captured by a category, abstractness is much more difficult to define implementationally. For example, is "intelligent" more abstract than "independent"? A second problem is that the relative importance of features may change dynamically or as a function of interaction of features. In choosing a car, color, size and speed may be given a particular ordering of importance; in selecting a mode of transportation, these same features may be given a different ordering. Finally, as pointed out by data on remindings, it is not always the most abstract features that are responsible for a reminding; often, content features play a role in the similarity that is not accounted for by a strict abstractness metric. Of course, abstractness may play a role in the relaxed match process despite these problems; certainly, there seems to be an implicit notion that people have about what types of features may be important in analogy.

Controlling retrieval with goals

However, an alternative proposal is to use the context of the current goals being pursued as a method of weighting features for a match. Remindings, I claim, have more to do with *what* the person is attending to than to any a priori notions about abstractness. It is these processing goals that provide more weight to particular features and allow others to disappear from a match. I propose that goals play a role in retrieval by affecting what features are attended to most in a similarity match process. Particular features of the case will be attended to based upon the cognitive goals operating at the time.

Here's an example of how goals may act to select features to attend to:

Input: Teletrack has a "no smoking" policy which means, in effect, that every section is a smoking section. The rule is disregarded often enough, by people seated in various sections, that no section can be guaranteed to be smoke-free. You argue that the management should allow smoking in some sections, if only to be sure that some sections can actually be non-smoking.

Suppose that you are engaged in an argument about your comment. Your goal is to buttress your claim that, basically, control is better than abolishment when disobedience is common. Supporting your claim involves retrieving another instance with this pattern; based on the abstract characterization of the claim, an analogous situation is recalled. You may be reminded, on this basis, of the legalization of heroin in England. That case supports the claim by identifying a circumstance where control of heroin use had fewer bad effects on non-participants than the outright banning of the substance.

Suppose instead that your goal is to plan a way to get your assertion implemented. In order to get your solution effected, you have to figure out a way to get the authority involved, Teletrack, to recognize that their control problem is worse now than it would be under the smoking section plan. Planning for implementing your solution selects problem features that are relevant to the management of the policy. You may be reminded, on this basis, of the pet policy in student housing. Pets were banned, but then many students got pets and broke the rules more and more brazenly, causing quite a bit of disturbance. In an attempt to manage the effects of the pets, the management made several buildings "no pets" and let the pet owners congregate in the others. This suggest a plan for the smoking policy: at Teletrack, smoke as much as possible, and encourage others to, in a way that exacerbates the problem for the management. They will be more likely then to see the benefit of control rather than prohibition.

Even if the plan does not seem a like good one, the point here is that the processing that goes on prior to a reminding may be much more involved than simply encoding input facts into a single representation. Instead, the features that

play a larger role in retrieval may be ones that are tied to particular processing goals active at the moment. From this, we could expect that retrieval of the new episodes later would depend on a congruency of cognitive purpose.For models where only one processing goal is ever present, the goals may be implicitly encoded in the process model and memory representation, compromising the ability to make claims about how cases are utilized in general.

The "goodness" of an analogy depends on what you are interested in; therefore the relevance of features changes dynamically as a result of the goals of a system. The goals may be modelled as connections to the input features that "gate" the activation of the features, in terms of multiplicative influences on the strength of the input features (as in Hinton's multiplicative links). The goal influence can turn down irrelevant and turn up relevant features in the match, resulting in the features relevant to the goal being the focus of the relaxed match. The reason for the separation of the goal links to the features from the input line architecture is to allow the same goals to function autonomously to the retrieval process, since they will also affect other subprocesses besides retrieval. Tasks like creating an explanation or learning can also be affected by the current processing goals. While the implementation of an abstractness metric through input line activation may be possible, that information has to be determined over many different processing contexts -- the abstractness of features may be learned over many different tasks. The gating connections from goals will provide activation based on information about what is relevant in the *current* processing context, which changes depending on the goals of the system. Thus, the two types of information about feature importance can be incorporated into the same network model.

Extensions to Learning

Cognitive goals may affect other processes besides retrieval, operating to provide a context for saliency of features. The tendency to treat all features as equal, without regard to how particular features may be attended to in certain cognitive contexts, is a major problem in models of learning. For example, distributed models of learning as a class consider all features present as potentially equally important to the rule being learned. Consequently, it takes many trials to determine which features are actually predictive rather than occurring occasionally. This approach hurts learning rates in two ways; first, it may hurt mathematically in the learning algorithms to include so many features that take many trials to be ruled out; and second, it ignores information available in utilizing a priori notions of what features are likely to be related to the rule.

In a learning system, treating all features equally will mean a lot of effort spent on features that have no importance to the rule learned. With the addition of a mechanism to select the features most likely to be relevant to the rule, one can save time by focusing on features that will pay off more quickly. For example, the current state of the network can provide a focus on particular input features based on what features have been given higher weights in previous learning. Consequently, by utilizing prior knowledge, one can shut off features that aren't a priori relevant to learning. Of course, as I have been arguing in this paper, this is not enough when the same network serves other tasks. Which features are relevant in a domain can't be set in an a priori way that will be true across all learning contexts. One needs a mechanism to affect the attention at the nodes *depending upon* the processing context. The goals can affect which features are attended to by interacting in a multiplicative fashion with activation on the input nodes. This method can incorporate changes in relevance due to the particular goals being pursued.

A particular method to control learning might be to focus on input values which have small outgoing weights, in a sense betting that the information to be

learned will have to do with the features already shown to be relevant in previous learning. Of course, when the same network is used for other tasks, then a gating method will be required as proposed for the retrieval model, to allow the current system goals to directly impact which features are attended to. This problem of selecting features to attend to in learning has begun to receive some interest recently. M. Mozer (personal communication, March 1988) has begun studying the problem using a bootstrapping learning procedure that attenuates input values with small weights. The results are suggestive: with noisy inputs, the network decides to ignore the noise inputs *before* solving the mapping, and in the case of redundant inputs, the attentional mechanism selects one and shuts out the redundant feature.

It seems that people often have notions of what features *may* be important to learning in a particular domain. For example, in learning a new video game, the actions effective in killing enemies tend to be predictable from common-sense notions of physical causality (e.g., you have to be in a linear position with the target for a shot to hit it). To the extent that the actual game mechanics violate these notions, they may be increasingly hard for people to learn. In fact, the less face validity to the rule, the more often people may fail to perceive its presence among the possible factors. This is a drawback to the process of using default expectations for the relevancy of features; however, it may be more than compensated for by the ability to quickly detect the operation of more obvious factors. By attending to the seemingly related features, the learning process may be "smartened up" and sped up, at the cost of discovering nonintuitive or novel connections in the data. Within human systems at least, the gain from attention in speed of learning may more than make up for missing counterintuitive or unusual patterns. Utilizing cognitive goals to affect the attention paid to features based upon their inferable connections seems a promising approach for models of learning as well as retrieval. I think the combination of network strengths (content addressing) and a controlling mechanism (attention based on goals or previous learning) is a promising method to "smarten up" the behavior of networks in retrieval and speed up the learning process.

References

Chandrasekaran, B. (1987). Towards a functional architecture for intelligence based on generic information processing tasks. *Proceedings of the Tenth IJCAI*, Milan, Italy.

Dyer, M. G. (1983). *In-depth understanding: A computer model of integrated processing for narrative comprehension.* Cambridge, MA: MIT Press.

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

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.

McKoon, G., Ratcliff, R., & Seifert, C. M. (1988). Making the connection: Generalized knowledge structures in story understanding. Unpublished Manuscript.

Rumelhart, D. E. (1988). Towards a microstructural account of human reasoning. Unpublished manuscript.

Schank, R. C. (1982). Dynamic memory: A theory of reminding and learning in computers and people. New York:Cambridge University Press.

Seifert, C. M., McKoon, G., Abelson, R. P., & Ratcliff, R. (1985). Memory connections between thematically similar episodes. *Journal of Experimental Psychology: Human Learning and Memory, 12* (2), 220-231.

The Role of Mapping in Analogical Transfer[1]

Hong S. Shinn
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332, U.S.A.

Abstract

This paper aims to provide a view of the role of analogical mapping in the entire process of analogical problem solving. In many models, analogical mapping is responsible for identifying the analogy between two problems by considering structural and semantic similarities. However, given a non-trivial analogy problem, success of mapping does not always guarantee successful transfer of analogy. In fact, there exist many analogy problems, which succeed on analogical mapping but which fail on analogical transfer. While a potential mapping between problems can be generated, that mapping might not be justifiable until transfer from one problem to another is attempted.

We present our analogical mapping method and show how it works for inter-domain and intra-domain analogies. We demonstrate several analogy problems in which a mapping can be generated that cannot be transferred. We also compare our method to two general mapping mechanisms, SME and ACME, and show that it performs at least as well, and sometimes better than either of those methods.

Key Words: analogical problem solving, analogical mapping.

1 Introduction

This paper aims to provide a view of the role of analogical mapping in the entire process of analogical problem solving. Analogical problem solving contains at least the following components [Shi88,CM85,HT88]: retrieval of a plausibly analogous case, analogical mapping, and analogical transfer. The step of analogical transfer may involve modification of a previous solution and justification of the result obtained before the result is transferred [Shi88].

In many models, analogical mapping is responsible for identifying the analogy between two problems by considering structural and semantic similarities. However, given a non-trivial analogy problem, analogical mapping by itself does not always guarantee that an analogy will be successful. While it can produce a potential mapping between problems, a mapping might not be justifiable until transfer from one problem to another is attempted.

In this paper, we illustrate the role of mapping in analogical problem solving and we present a hierarchical method of analogical mapping that is based primarily on similarity of structures. The method uses relatively little semantic information. Instead, it relies on the analogical transfer step to determine the merit of a potential analogical mapping.

We show how our method works for inter-domain and intra-domain analogies. We demonstrate several analogy problems in which a mapping can be generated that cannot be transferred. We also compare our method to two general mapping mechanisms, Structure-Mapping Engine (SME) [FFG86] and Analogical Constraint Mapping Engine (ACME) [HT88], and show that it performs at least as well, and sometimes better than either of those methods.

[1]This research has been supported in part by the Army Research Institute under Contract No. MDA-903-86-C-173, is currently supported in part by NSF under Grant No. IST-8608362, and in part by Lockheed AI Center under Grant No. DTD 09-25-87.

2 Analogical Mapping Algorithm

Before introducing our algorithm, we need to clarify the problem of analogical mapping. Polya [Pol54] views "analogy" as a systematic correspondence between two systems preserving certain relations. For his basic type of analogy, Polya defines analogy as "similarity of relations", where relations are similar if they are governed by the same laws. He illustrates this with an example: the multiplication of numbers *multiply(x,y)* is analogous to the addition of numbers *add(x,y)* in the sense that both multiplication and addition are commutative. In other words, two relations *multiply(a,b)* and *add(a,b)* are similar because they are governed by the same commutative law: *equals[OP(a,b),OP(b,a)]*. Interpreting Polya's definition of similarity in analogical problem solving, "similarity" in problems is what leads to similar effects on their solutions. The problem here is that, without knowing beforehand what the similarity's effect will be on the solution to the target problem, we must find that similarity which can be used in deriving the solution. Thus, what analogical mapping does is to find the most probable similarity candidates before transfer of knowledge from source to target is attempted.

Our analogical mapping algorithm follows Gentner's systematicity principle ([Gen83], p. 163) in that it transfers "a system of connected knowledge, not a mere assortment of independent facts". In other words, during mapping between structures, even the highest order predicates may not be mapped separately from their lower level entities.

In dealing with similarity, however, we do not accept Gentner's entire theory of structure mapping. In our mapping scheme, two relations which are structurally similar (i.e., the current partonomic roles in both structures are the same) will not be thrown out. For example, according to Gentner, two relations *equals[multiply(a,b),multiply(b,a)]* and *equals[add(a,b),add(b,a)]* are not mappable to each other because the highest level predicates are identical (i.e., *equals*), but not the lower level predicates (i.e., *add* and *multiply*). On the other hand, in our scheme, these are mappable because their high order predicates are the same while the low level predicates "add" and "multiply" are structurally similar due to their similar roles in the whole relations. Burstein [Bur86] demonstrates with his system CARL the necessity of mapping between nonidentical relations, criticizing Gentner's structure mapping which fails on this kind of similarity.

Another characteristic of our mapping scheme is hierarchical mapping. This is frequently used when problems are represented in hierarchical structure. In fact, analogy between problems usually exists at an abstract level. Thus, mapping starts at the highest level first and proceeds to the next lower level and so on until analogy breaks down.

In our scheme, the entire mapping process is a recursive application of a two-step hierarchical mapping: first map the two problem structures systematically under structural similarity and then decompose them into the next lower level structures (see Figure 1). Structural similarity is found not only in physical structures but also in functional structures. Functional structures are described by functional objects and relations such as functions, purposes, goals, constraints, conditions, and states. For example, *an air conditioner is like an electric fan* because their top level functions are the same (i.e., excite-air). Another example of analogy is found between *society* and *organism* because they are similar in their functional organizations.

As a result of analogical mapping, an analogy map is generated for two cases showing correspondences between both relations and their objects. An analogy map represents a common structure between source and target structures with a binding list between source and target elements. The common structure represents a common problem schema which is used as a medium of transfer in analogical problem solving [Shi88,CM85]. For example, analogical mapping between *multiply(a,b)* and *add(a,b)* generates the analogy map as a common structure *OP(a,b)* with the binding list [(OP multiply add)] meaning that there is one binding *OP* and it binds to *multiply* in the source and to

Input: A source case and a target problem
Output: An analogy map (AMAP)
Algorithm:
Recursive application of two-step hierarchical mapping:
given two problem structures,

1. Map them systematically under structural similarity:
   identify components whose partonomic roles in both structures are the same;
   map components as specifically as possible under the current AMAP;
   if mappable
   then add correspondences between components to AMAP
   else return AMAP
2. Hierarchical refinement:
   decompose the current level into the next lower level structures;
   pair them in the same partonomic roles

Figure 1: Analogical Mapping Algorithm

add in the target.

3 Related Work

Gentner's structure mapping theory with its implementation, SME [FFG86], demonstrates the importance of systematicity in interpreting an analogy. But, it is often criticized because of its syntactic approach.

Many recent models consider semantic and pragmatic characteristics of analogy as well as syntactic information to guide analogical mapping. For instance, Burstein [Bur86] introduces some top-down constraints on relations and primarily relates objects in terms of their functional roles in analogical mapping. Winston's mapping is driven by importance-dominated matching [Win80,Win82]; importance is mainly determined by causal relations in the situations.

Holyoak and Thagard's mapping theory [HT88] attempts to take into account all three dimensions of analogy: syntax, semantics and pragmatics. Their program called ACME computes an analogical map by means of constraint-satisfaction based on five heuristic constraints[2]: logical compatibility, uniqueness, relational consistency, semantic similarity, and role identity. Semantic and pragmatic information help to constrain the search for the most plausible mapping. But, the problem with this approach is that there exist many analogy problems on which such heuristics do not work (an example will be shown in Section 5.1).

Our mapping algorithm is similar to SME in that both enforce systematicity (as shown in the previous section), but different in that ours maps predicates under similarity by functional roles while SME maps under identity. Ours is also similar to Burstein's and ACME in that it maps components by considering part-whole relationships. However, unlike ACME and Winston's, much of the semantic information is not explored during mapping. Rather, it will be checked when the knowledge to be transferred is justified in the transfer step.

[2]Several of these constraints were renamed in a later version of ACME [personal communication with Keith J. Holyoak, May 1988].

4 Applications of Analogical Mapping

Analogical mapping is a step of predicting a plausible analogy, which will be tried for transfer. During the actual transfer attempt, mapping results are filtered considering semantic similarity. The following applications show how these processes are performed.

The first two applications of our mapping algorithm rely on similarity in physical structures: Section 5.1 shows analogy examples between different domains, while Section 5.2 compares analogies within the same domain. These examples are also used to compare our algorithm to two general analogical mapping mechanisms, SME and ACME. In Section 5.3, an application from the JULIA project shows an example in functional structures.

4.1 Inter-Domain Examples

Applying our analogical mapping algorithm, let's solve the problem $\frac{d}{dx}[\sin x - \ln x]$ using the following case:

Problem: $\int [e^x + 1]\, dx$
Solution: $e^x + x + C$
Reasoning steps: $\int [e^x + 1]\, dx \Longrightarrow \int e^x\, dx + \int 1\, dx \Longrightarrow e^x + x + C$

When the mapping algorithm, in the first cycle, is applied to the top level structures (i.e., $\frac{d}{dx}[\sin x - \ln x]$ and $\int [e^x + 1]\, dx$), it successfully produces the analogy map

$$\mathcal{F}[f(x)\, OP\, g(x)]$$

with bindings $[(\mathcal{F}\ \frac{d}{dx}\ \int)\ (\mathrm{OP}\ -\ +)\ (f(x)\ \sin x\ e^x)\ (g(x)\ \ln x\ 1)]$. Since the first reasoning step of the source case predicts the following analogy (in an abstract form):

$$\mathcal{F}[f(x)\, OP\, g(x)] = \mathcal{F}[f(x)]\, OP\, \mathcal{F}[g(x)]$$

the target problem reduces as follows:

$$\frac{d}{dx}[\sin x - \ln x] = \frac{d}{dx}\sin x - \frac{d}{dx}\ln x$$

In the next cycle, the mapping between the next lower level structures $\int e^x\, dx$ and $\frac{d}{dx}\sin x$ succeeds, but analogical transfer between these two fails. This is the level where the analogy breaks down and the mapping process halts. Thus, the analogy between the above two cases resides only at the top level. This example shows the utility of hierarchical mapping in identifying analogy, since the analogy at higher levels of abstraction can be used even though there does not exist a complete analogy.

Consider another problem

$$e^{2x+3}$$

using the same source case. It is similar to the first example in that mapping predicts

$$e^{2x+3} = e^{2x} + e^3$$

However, this hypothesis is not correct; the correct transformation is $e^{2x+3} = e^{2x} * e^3$. This example shows that successful analogical mapping may not guarantee the existence of analogy when semantic similarity is missing. The semantic similarity is checked using reasonings similar to those of the source case. This is done during the process of analogical transfer to justify the hypothesized analogy. (See [Shi88] for more discussion of the justification problem.)

Let us apply SME and ACME to the first analogy problem:

Target: $\frac{d}{dx}[\sin x - \ln x]$
Source: $\int [e^x + 1]\, dx$

SME would fail to recognize this analogy because the two high level predicates $\frac{d}{dx}$ and $\int$ are not identical. This example shows that structural mapping under predicate identity is too strong. In ACME, the logical compatibility requires the second arguments $\ln x$ and 1 to be the same logical kind (e.g., n-place predicates to n-place predicates, constants to constants). Because this constraint precludes a mapping between the one-place predicate $\ln x$ and the constant 1, ACME would also fail on this analogy. This case suggests that semantic and pragmatic information should be used cautiously because of their heuristic nature.

4.2 Intra-Domain Examples

Given a problem

$$\int \frac{1}{\sqrt{3 - y^2}}\, dy$$

consider analogical mapping problems with each of the following three cases.

Case 1:

Problem: $\int \frac{1}{\sqrt{3+y^2}}\, dy$
Solution: $\sinh^{-1} \frac{y}{\sqrt{3}} + C$

Case 2:

Problem: $\int \frac{1}{\sqrt{1-x^2}}\, dx$
Solution: $\sin^{-1} x + C$

Case 3:

Problem: $\int \frac{1}{\sqrt{5-z^2}}\, dz$
Solution: $\sin^{-1} \frac{z}{\sqrt{5}} + C$

All three mappings succeed with our mapping algorithm because the three cases are all structurally similar to the target problem.

In the first case, analogy transfer from case 1 to the target problem is not possible (because the previous reasoning of case 1 is not applicable to the target problem). In the second case, transfer from the source case is not possible until some modification is performed. That is, in order to apply the solution of case 2 to the target problem, the form $\sqrt{a - z^2}$ embedded in the target problem needs be transformed to the form $\sqrt{1 - x^2}$ in case 2. In the third case, the source solution can be transferred to the target domain so the target solution will be $\sin^{-1} \frac{y}{\sqrt{3}} + C$.

The success of analogical mapping leads directly to analogy transfer in the third example. The first example shows, however, that the success of mapping may not guarantee successful analogy transfer. (It only predicts a possibility of transfer which should subsequently be verified.) Furthermore, the second example shows that even when analogical mapping eventually leads to analogy transfer, successful analogical mapping may not directly dictate what is to be transferred from the source case to the target problem. (It may only hint at what is to be transformed in order to reach a transferable state.) So, the role of analogical mapping is to identify a plausible analogy based on known similarity before transfer of analogical knowledge from the source case to the new problem is attempted [Shi88].

Note that SME and ACME are similar to our mapping algorithm in that they will come up with successful mappings with all three cases. This shows that, even when ACME considers semantic and pragmatic accounts, it is not able to distinguish analogies which lead to analogical transfer (i.e., case 3) from analogies which do not (i.e., cases 1 and 2). In other words, ACME is not more powerful than SME and ours in dealing with these three.

4.3 An Application in JULIA

Our analogical mapping mechanism is part of the case-based reasoner [Shi88] in JULIA, an intelligent caterer's advisory system [CK86,Kol87]. Each problem case in JULIA has problem and solution parts. The problem part describes its problem functionally in terms of goals and constraints while the solution part contains a solution plan and the reasoning history.

Since problem cases are represented in a hierarchical structure, JULIA maps the top level problem structures first. It tries to identify similarity between two functional structures. In a frame-based representation, it is straightforward to identify the same functional components (e.g., goals and constraints). JULIA starts mapping with goals between problems: if the goals fully match, the mapping proceeds to constraints; in case of a partial match, which means some goals match but others do not, only the matched goals will be considered for possible transfer; otherwise, the mapping fails. Mapping then proceeds to constraints on only the matched goals to establish correspondences between them. For example, JULIA would consider two cost constraints LOW-COST and INEXPENSIVE[3] similar, because they are functionally the same in that they both constrain the cost. Then, should LOW-COST and EXPENSIVE be considered similar, too? JULIA views that they, too, are functionally similar due to the same reason. However, these do not have as much in common semantically as LOW-COST and INEXPENSIVE.

This problem will be resolved during actual transfer and there the degree of semantic similarity determines the degree of learning involved. Suppose the source case made the following inference during its problem solving:

If c-cost(LOW-COST)
then c-ingredient-cost(LOW-COST) and c-cooking(LOW-COST)
because cost of dish is cost of ingredients plus cooking cost

Then, during analogical transfer, JULIA will try to transfer the previous inference rule with the similar concept INEXPENSIVE using its justification ("because") clause. In other words, JULIA hypothesizes a rule substituting LOW-COST in the rule for INEXPENSIVE, seeing if the justification previously used is similarly applicable. Since the justification also holds for the target case, the new rule will be transferred:

If c-cost(INEXPENSIVE)
then c-ingredient-cost(INEXPENSIVE) and c-cooking(INEXPENSIVE)
because cost of dish is cost of ingredients plus cooking cost

However, if it were EXPENSIVE, the similar inference may not be true because not every ingredient needs to be expensive to make a dish expensive.

[3] INEXPENSIVE ranges from low cost to moderate cost so that its meaning is slightly broader than that of LOW-COST.

5 Summary and Conclusions

We have shown that successful mapping may not guarantee successful transfer of analogy. Analogical mapping only predicts a possibility of transfer which should subsequently be verified. Even when analogical mapping eventually leads to analogy transfer, successful analogical mapping may not directly dictate what is to be transferred from the source case to the target problem.

An analogical mapping algorithm has been introduced as a recursive application of two-step hierarchical mapping: first map the two problem structures systematically under structural similarity and then decompose them into the next lower level structures. Structural similarity is identified during this mapping process, while semantic similarity is checked during analogical transfer. These two processes together guarantee the correctness of analogy transfer.

Acknowledgments

This work could not have been done without the support and guidance of Janet L. Kolodner. I would like to thank my fellow-students Patsy L. Holmes, David Wood, Mark A. Graves, Joel Martin, and Mike Redmond for useful comments and discussion on earlier versions of this paper. I would like to especially thank Keith J. Holyoak for his valuable comments.

References

[Bur86] M.H. Burstein. Concept formation by incremental analogical reasoning and debugging. In R.S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, Kaufmann, Los Altos, CA, 1986.

[CK86] R.E. Cullingford and J.L. Kolodner. Interactive advice giving. In *Proceedings of the 1986 IEEE International Conference on Systems, Man, and Cybernetics*, 1986.

[CM85] J.G. Carbonell and S. Minton. Metaphor and commonsense reasoning. In J.R. Hobbs and R.C. Moore, editors, *Formal Theories of the Commonsense World*, Ablex, Norwood, NJ, 1985.

[FFG86] B. Falkenhainer, K.D. Forbus, and D. Gentner. The structure-mapping engine. In *Proc. AAAI-86*, 1986.

[Gen83] D. Gentner. Structure-mapping: a theoretical framework for analogy. *Cognitive Science*, 7:155–170, 1983.

[HT88] K.J. Holyoak and P. Thagard. Analogical mapping by constraint satisfaction. 1988. Unpublished manuscript, Department of Psychology, UCLA.

[Kol87] J.L. Kolodner. Capitalizing on failure through cased-based inference. In *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, Seattle, Washington, July 1987.

[Pol54] G. Polya. *Mathematics and Plausible Reasoning: Induction and Analogy in Mathematics.* Volume 1, Princeton University Press, Princeton, NJ, 1954.

[Shi88] H.S. Shinn. Abstractional analogy: a model of analogical reasoning. In *Proceedings of the DARPA Workshop on Case-Based Reasoning*, May 1988.

[Win80] P.H. Winston. Learning and reasoning by analogy. *Comm. ACM*, 23(12):689–703, 1980.

[Win82] P.H. Winston. Learning new principles from precedents and exercises. *Artificial Intelligence*, 19:321–350, 1982.

Spatial attention and subitizing:
An investigation of the FINST hypothesis

Lana Trick and Zenon Pylyshyn

University of Western Ontario

How are visual items counted? It seems obvious. You find an item, augment a counter and then mark the item as counted. When all of the items have been marked you say the counter value. If this were true, counting latency should be a simple function of the number of items. Reaction time should increase a constant amount with every additional item after one. This is not the case, however. Research dating from the 1870's on has shown that the reaction time increase is not uniform. When there are a small number of items, the slope is shallow, at most one tenth second. Thereafter, the slope increases to one third second. The subitizing range, usually 1-4 items, is the range in which the slope is shallow. The subitizing process is rapid, accurate, and effortless. Above the subitizing range, from 5 on, is the counting range. Counting is a slower, more effortful and error prone process.

Why do subitizing and counting differ? First, a digression is required. It is well established that items differing from distractors by a primitive feature such as color, orientation etc., "pop out" in search tasks (Treisman and Gellade, 1980). For example, subjects detect the presence of a red item in a field of blue items in a time independent of the number of items in the display. In contrast, when an item is a different conjunction of features (eg. a red square in a field of red triangles and blue squares) or a subset (eg. an O in a field of Q's), search time depends on the total number of items. The standard interpretation is that a limited capacity spatial processor combines

features; an attentional "spotlight" is moved from location to location in order to form an integrated object representation. However, when an item differs from distractors by virtue of a simple feature attention is not required. Features are computed by parallel preattentive mechanisms, and thus "odd man out locations" pop out. Pylyshyn (1987;1986) proposed that a small number of pop out locations can be assigned internal reference tokens, FINSTS. FINSTS remain assigned to a feature cluster though its retinal position changes. Thus, FINSTS permit the construction of geostable spatiotopic representations from the retinotopic output of low level feature extraction processes. According to Pylyshyn, only FINSTED locations can be accessed by the attentive processes that combine features and compute abstract spatial relations (Ullman, 1985). What does this have to do with subitizing and counting? I would like to argue that subitizing is parasitic this preattentive "FINSTING" mechanism. Therefore, subitizing should not be possible when attention is required to disambiguate the items that need to be counted from distractor items. If the items to be counted differ from the distractors by a simple feature, however, subitizing should occur. Counting, on the other hand, a process that by its nature involves moving a spatial processor through a representation (Ullman, 1985), should be unaffected by the type of distractor.

METHOD

Subjects

Nine graduate students, five female, participated for payments of twenty dollars. Each subject participated in every experimental condition.

Apparatus and Materials

Vocal response latencies were measured using a Gerbrands G1341 voice activated relay, while an Apple II+ computer was used to display the

stimuli and record the data.

The stimuli were displays of up to 15 white letters on a black background. At most the letter display could cover a 5.97 by 4.2 degree area of visual angle. Letters each occupied .36 by .21 degrees and the minimal distance between letters was .73 horizontal and .36 vertical degrees when the subject was seated 110 cm from the video screen.

Subjects were required to count letter O's in a background of distractor letters. There were, in most cases, 1 to 8 O's and either 0, 2 or 4 distractor letters. Catch trials were also included in which there were no O's or 9 O's, or 1, 3, or 7 distractor letters. In the Pop out condition the distractor letters were X's; pilot studies showed that O's clearly "pop out" of X's in search tasks. In the Attentive search condition the distractor letters were Q's; in accord with Treisman (1985) we found O's do not "pop out" of Q's.

Procedure

The experiment was run in two sessions. In one session subjects were required to count O's in a background of X's (Pop out condition) whereas in the other subjects were required to count O's in a background of Q's (Attentive search condition). See figure 1. Session order was counterbalanced.

Each trial had four stages. First, during the 512 msec pre-trial interval, the screen went white and subjects were required to fixate on a central green dot. The computer then beeped to signal the start of the trial. After 256 msec, the letter display appeared and remained on until subjects made a vocal response. Third, as soon as the timer registered the response the display disappeared and the screen turned white. Finally,

after 512 msec subjects were prompted to type in the number that they had said. Accuracy feedback was given.

At the beginning of each session subjects were given 60 practise trials.

RESULTS AND DISCUSSION

To summarize, the ability to subitize seems to be eradicated when subjects have to attentively search for the target letters; when subjects count O's in a background of Q's they no longer seem capable of subitizing. The slope of the reaction time function remained constant throughout the 1-8 range. When the targets pop out from the background, however, the ability to subitize is spared. Thus, when subjects count O's in a background of X's there is evidence of subitizing once more; the slope of the reaction time function in the 1-4 range is different than it is in in the 5-8 range.

Reaction time analysis

Averaged data. As can be seen from figure 2, both the number of O's and the number of distractors yielded effects on reaction time. In fact, analysis of variance revealed all main effects and interactions to be significant. Reaction times increased as the number of O's in the display increased($F(7,56)=515.5, p<.01$). Overall, however, the increases were greater in the Attentive search session than the Pop out session($F(7,56)=48.1, p<.01$). Further, number of distractors also affected reaction time; the more distractors the longer it took to count($F(2,16)=1392.2, p<.01$). Once again, though, number of distractors had a greater effect when the distractors were Q's then when they were X's($F(14,112)=7.8, p<.01$). Finally, the deleterious effect of distractors

varied according to the number of O's and the type of distractors($F(14,112)=2.52, p<.01$).

Subitizing ranges were determined empirically through trend analysis. First, in order to determine if subitizing occured, trend analysis was performed on the entire range (1-8 O's) to find out if significant trends beyond the linear emerged. If the reaction time function showed no significant deviation from linearity then it was assumed that subitizing did not occur. However, if there were significant deviations from linearity it was necessary to find out where the trend emerged and if it was in the right direction. The point at which the reaction time function began to show significant positive quadratic trends was judged to be the boundary of the subitizing range.

When there were 2 or 4 Q's in the display with the O's no significant nonlinear trends in the reaction time emerged, although in both cases the linear trends were highly significant(linear $F(1,64)=851.1, p<.01$; nonlinear deviation $F(6,64)<1$; linear $F(1,64)=884.7, p<.01$; nonlinear deviation $F(6,64)<1$ for 2 and 4 distractors respectively). In contrast, when there were 2 or 4 X's in the display, subitizing was observed. Significant linear trends emerged but also significant deviations from linearity and quadratic trends(nonlinear deviation $F(6,64)=2.29, p<.05$; quadratic $F(1,64)=9.8, p<.05$; nonlinear deviation $F(6,64)=2.28, p<.05$; quadratic $F(1,64)=11.23$, $p<.01$ for 2 and 4 distractors respectively). In the two distractor condition the quadratic trend emerged at 5 whereas in the four distractor condition the quadratic trend didn't emerge until 6. Thus, subjects seem capable of subitizing up to 4 O's when there were 2 X's and up to 5 O's when there were 4 X's. Nonetheless, the results support the idea that subitizing items in a background is possible only when the

items "pop out" of the background items.

As expected, subitizing was clearly shown whenever there were no distractors. In both the Pop out and Attentive search conditions there were linear and higher order trends (nonlinear $F(6,64)=6.1$, $p<.01$; quadratic $F(1,64)=24.0$, $p<.01$; nonlinear $F(6,64)=4.73$, $p<.01$; quadratic $F(1,64)=11.8$, $p<.01$ for Pop out and Attentive search respectively). In both cases the quadratic trend emerged at 5, indicating that subjects could subitize up to 4 O's when there were no distractors.

Slopes of the functions were revealed through regression analysis. (See table 1). For subitizing, in the pop out condition there seemed to be a steady increase in the slope as the number of distractors increased, consistent with the notion of the cost of filtering(Treisman, Kahnemann and Burkell, 1983). The only significant slope differences were between no distractor and four distractor condition however; in the other cases estimated slopes fell within each other's 95% confidence interval. (For simplicity in the analysis subjects were assumed capable only of subitizing to 4 in the four distractor condition. Later analyses revealed that most subjects subitize to 4 in this condition).

Surprisingly, when the subitizing slopes of the two no distractors conditions were compared, there were significant differences. The subitizing slope was significantly greater when subjects had no distractors in the Q distractor session than when there were no distractors in the X distractor session.

As expected, few systematic trends in the counting (5-8) range slopes. In all cases, counting slopes fell within each other's 95% confidence intervals. Thus, the type of distractor had little influence on the

counting slope.

Individual analysis. Given that averaging across individuals might obscure subtle changes in slope, trend analyses were also performed on individual datasets. Results were consistent with findings from the averaged data. (See table 2). None of the subjects showed non-linear trends counting O's in a field of Q distractors. In contrast, most showed non-linear trends when counting O's in X's (Chi square(3)=14.0,$p<.01$).

There are individual differences in where the quadratic trends emerge, however; some people seem subitize larger numbers than others. Although the majority of the subjects subitize to 4, there is some variability, and what's more, the greater the number of distractors the more variability there is.

Error analysis

Subjects made few errors. Nonetheless, condition, number of items, and number of distractors all had effects on the error rate. (See figure 3). Subjects tended to make more errors when counting in the Attentive search condition($F(1,8)=8.9, p<.05$). Also, the probability of error increased with the number of O's($F(7,56)=5.8$, $p<.01$). Distractors also affected error rate($F(2,16)=4.06$, $p<.05$).

Conclusion

As predicted, subitizing was not possible when attention was required to disambiguate items from the distractors but was in evidence otherwise. Subitizing seems to rely on preattentive mechanisms. Counting seems to require localized attention, however; thus distractor type had little effect on slopes.

References

Pylyshyn, Z. (1987). The role of location indexes in spatial perception: A sketch of the FINST Spatial-index Model. Cogmem, 23, University of Western Ontario.

Pylyshyn, Z. and Storm, R. (1986). Tracking of multiple independent targets: Evidence for both serial and parallel stages. Cogmem, 22, University of Western Ontario.

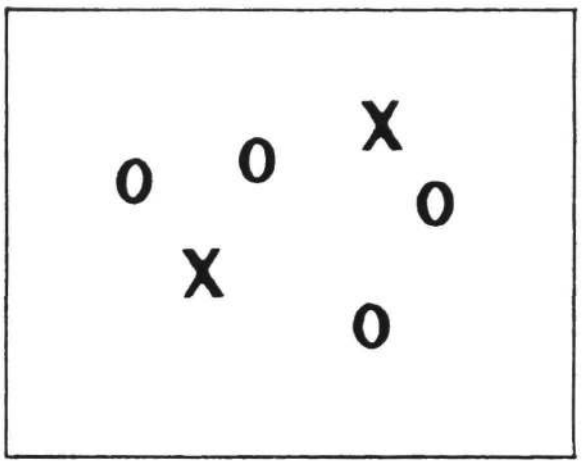
Treisman, A., and Gellade, G. (1980). A feature integration theory of attention. Cognitive Psychology, 12, 97-136.

Treisman, A., Kahneman, D., and Burkell, J. (1983). Perceptual objects and the cost of filtering. Perception and Psychophysics, 12, 527-532.

Treisman, A. (1985). Preattentive processing in vision. Computer vision, graphics and image processing, 31, 156-177.

Ullman, S. (1984). Visual routines. Cognition, 18, 97-159.

Pop out Condition
(2 distractors)



Attentive search Condition
(2 distractors)

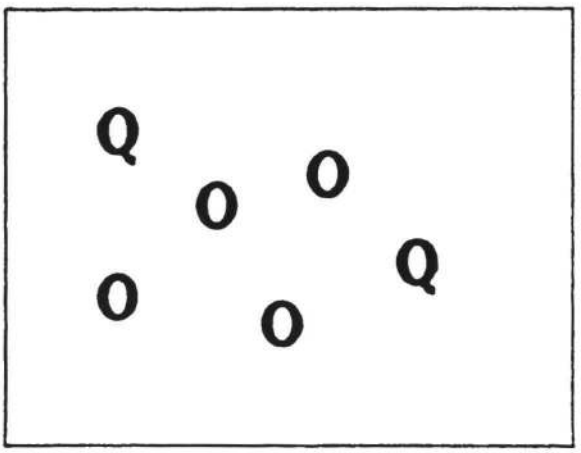


Figure 1. Counting Displays

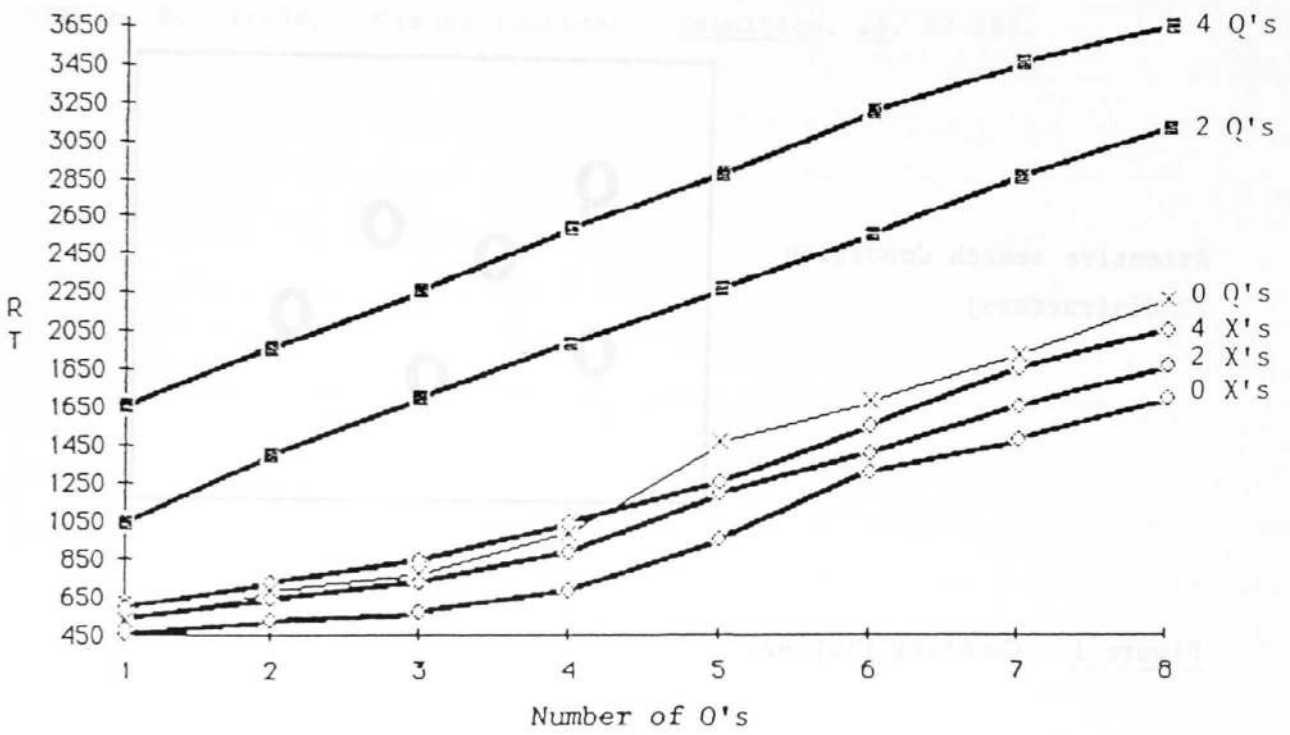
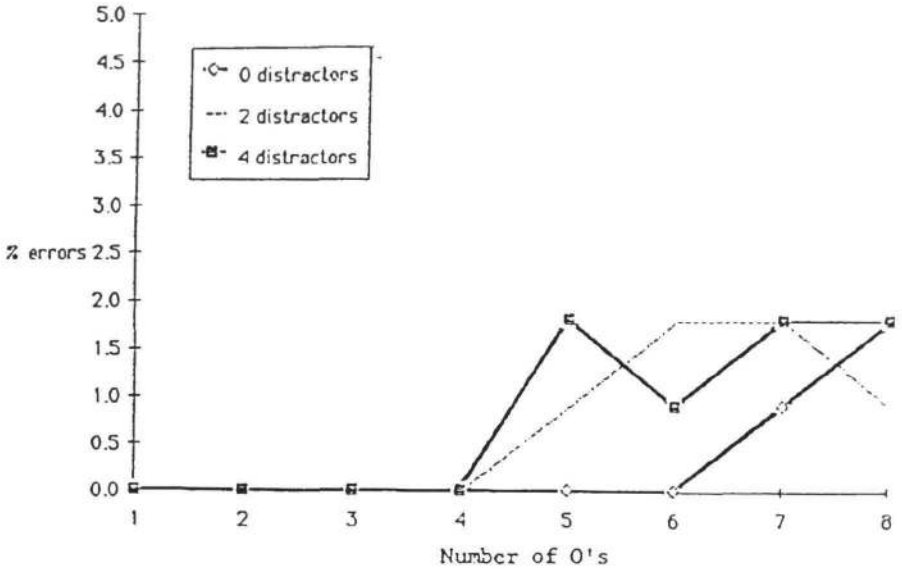
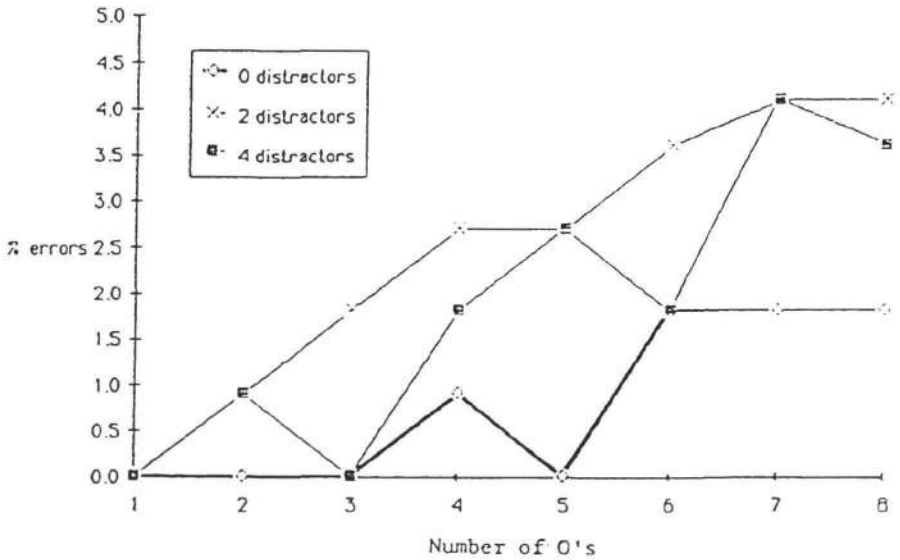


Figure 2. Counting latency averaged across subjects.



Pop out condition: Counting O's in the background of X's



Attentive search condition: Counting O's in a background of Q's

Figure 3. Average number of errors in counting

Table 1

Regression analysis of averaged counting latencies

	Slope	95% C.I.	R
SUBITIZING RANGE (1-4)			
Pop out condition: Counting O's in X's			
0 distractors	71.3	49 - 93	.75
2 distractors	115.2	85 - 146	.80
4 distractors	143.9	108 - 180	.81
Attentive search: Counting O's in Q's			
0 distractors	148.4	123 - 173	.90
2 distractors	--		
4 distractors	--		
COUNTING RANGE (5-8)			
Pop out condition: Counting O's in X's			
0 distractors	242.7	179 - 306	.80
2 distractors	227.4	164 - 291	.78
4 distractors	272.4	208 - 337	.83
Attentive search: Counting O's in Q's			
0 distractors	252.5	194 - 310	.83
2 distractors*	294.1	275 - 313	.96
4 distractors*	293.5	274 - 313	.96

*Because there was no evidence of subitizing in the Attentive search condition with 2 or 4 distractors the counting range is considered to be 1-8.

Table 2

Trend analysis of individual datasets

Number of subjects showing non-linear, quadratic trends in RT

	Pop out condition	Attentive search condition
0 distractors	9/9	7/9
2 distractors	7/9	0/9
4 distractors	7/9	0/9

Individual breakdown of subitizing range

POP OUT CONDITION	0 distractors	2 distractors	4 distractors
Total subjects	(N=9)	(N=7)	(N=7)
No. who subitize to 3	2	1	1
No. who subitize to 4	7	5	3
No. who subitize to 5		1	2
No. who subitize to 6			1

ATTENTIVE SEARCH	0 distractors
Total subjects	(N=7)
No. who subitize to 3	1
No. who subitize to 4	6

A Computational Model of Reactive Depression

Charles Webster[1,2], MD, Richard M Glass, MD[3], Gordon Banks, PhD, MD[1]

[1]Decision Systems Lab, Dept. of Neurology, U of Pittsburgh
[2]Lab for Computational Linguistics, Dept. of Philosophy, CMU
[3]Dept. of Psychiatry, U of Chicago

Introduction

We propose a learning model of normal reactive depression and describe DEP (Depression Emulation Program), a computer simulation of selected aspects of information processing in depression. Behaviors we address include the role of stable, internal, and global failures in triggering depression, increased objectivity, the cognitive loop, fluctuations of self-describing adjectives, and decreased motivation. These behaviors are consistent with the hypothesis that reactive depression involves controlled processing of the self-schema in order to learn new automatic emotional processes for directing attention to goals and plans.

Occasional mild depression may be an adaptive response to a changing environment (Klerman, 1987). Emotions narrow attention to personally relevant inferences (de Sousa, 1987). In natural language understanding, emotional context provides powerful constraints for understanding the goals and plans of characters in narrative (Dyer, 1983). In cognitive psychology, emotions have been hypothesized to motivate the selection of situationally and personally relevant goals (Simon, 1967). We theorize that the nondepressed person goes about his daily activities with largely automatic emotional responses directing attention to goals and plans. A depressed person, however, seems to require much effort in order to carry out the same activities. After a failure (losing a job, for example), a person may have to adopt new automatic strategies, and reactive depression may actually facilitate the discarding of obsolete behaviors and the formation of new ones. During this reformation, automatic processing of the self-schema may be suspended in favor of more slow, effortful, and sequential processing.

DEP is a program embodying a computational learning theory of adaptive reactive depression (transient depressed mood following an environmental event; all references to depression herein will be to this specific type). In the nondepressed state, DEP directs attention rapidly and effortlessly (automatic processing), without environmental feedback. In the depressed state, DEP directs attention slowly and effortfully (controlled processing), while environmental feedback modifies attention-directing emotional responses. DEP models attention direction using a connectionist discrimination network, DEP's analog of the self-schema. Automatic network descent is achieved when each node in the descent path has sufficient activation. In controlled descent, insufficient activation leads to slow sequential search for a correct descent path. When a correct path is found, DEP's weights are adjusted to increase the activation level of the path in response to future inputs. Thus, controlled descent of DEP's self-schema is instrumental in the development of future correct automatic responses. DEP is not based on a theory of clinical depression. However, a model of "normal", reactive depression may be a useful intermediate step toward a computational model of "pathological" depression.

Automatic and Controlled Emotional Processing of the Self-Schema

We use two concepts from cognitive psychology in our model of depressive information processing: the self-schema (Marcus, 1980), and automatic and controlled processing (Schneider & Shiffrin, 1977; Shiffrin & Schneider, 1977; Schneider et al., 1984; Schneider & Detweiler, 1987). The self-schema is a hierarchical representational system for encoding, organizing, and retrieving declarative and procedural information about the self. It goes under a variety of names: self-concept (Shavelson, 1986), system concept (Carver & Scheier, 1986), self-schemata (Marcus, 1980), and self-schema (Coyne & Gotlib, 1983). We hypothesize that the self-schema can exist in an automatic (nondepressed) or controlled (depressed) state. An automatic process occurs as a "fast, parallel, effortless process that is not under the direct

subject control, and is responsible for the performance of well-developed skilled behavior." (Schneider et al., p. 1, 1984). Controlled processing is a "slow, generally serial, effortful, capacity-limited, subject-regulated processing mode that must be used to deal with novel or inconsistent information." (Schneider et al., p. 2, 1984). Learning new automatic processes requires controlled processing.

We conceive of the self-schema as a hierarchy of attributes and procedures enabling a person to select goals and plans specialized for important social roles (act attentive in class, take notes, act cultured at the opera, comment on the quality of the singing, etc.) based on stable self-attributes (degrees of intelligence, attractiveness, lovableness, etc.). "Motivated" (intensely and persistently goal-directed) behavior occurs when an individual automatically traverses the self-schema from the top to the bottom to arrive at a leaf node with executable plans. Figure 1 is a prototypical self-schema. Environmental features such as it being a workday and being hungry activate nodes in the self-schema. The weights between the upper nodes and the intermediate nodes encode self-attributes, such as "hardworking" and "ambitious". The terminal nodes activate plans like getting out of bed, going to work, and asking the boss to lunch. The self-schema is used like a discrimination network (Feigenbaum & Simon, 1984). For example, a person whose self-schema strongly encodes "hardworking" and "ambitious" self-descriptions will automatically go to work and ask the boss to lunch if it is a weekday lunchtime.

Deciding which arc to take during downward traversal of the self-schema requires an "emotional" reaction to plan outcomes (would the outcome make one happy, unhappy, frustrated, angry, etc.?). The path selected depends on outcomes of previously selected goals and plans in similar situations. Normally the traversal from the top to the bottom of the self-schema is automatic and self-schema behaves in a stereotypical manner, as in familiar social roles, effortlessly directing attention to self-relevant goals and plans, and creating default expectations for the results of actions. However, if the individual fails in an important social role, perhaps habitual behavior was not flexible enough, attention was misdirected, or wrong expectations were generated. Personal failure (for example, asking the boss to lunch and being told you are in danger of being fired) is normally a novel situation requiring a flexible response involving controlled processing. Counter-acting the failure may require controlled processing to block, change, or trigger automatic processes. Creation of new, less failure prone, automatic emotional processes requires controlled processing of the self-schema.

A major theory of depressive cognition can be understood in this light. According to the revised learned-helplessness model (Seligman, 1984), depression occurs when people explain negative personal outcomes in terms of *stable*, *internal*, and *global* factors. Failure usually requires specific attributions to trigger depression. In terms of our hierarchical self-schema and its role in focusing attention on appropriate goals and plans such a mechanism makes sense. We would only want to change such a representation if the failure is attributed to stable factors ("My performance will continue to disappoint the boss"), internal factors ("I could have prevented his dissatisfaction"), and to global factors ("I have similar problems in other areas

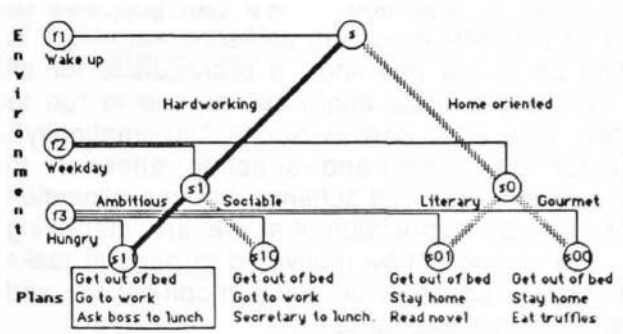


Figure 1: Prototypical Self-Schema

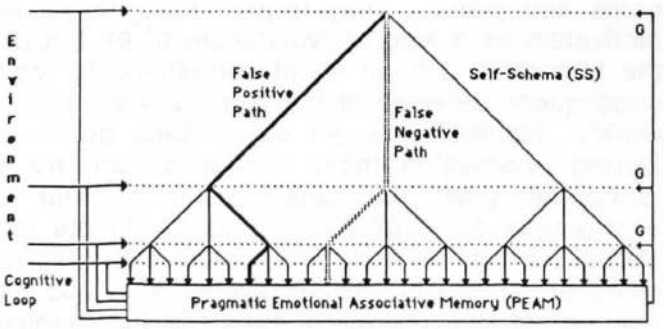


Figure 2: DEP Architecture

of my life"). To the degree that failure meets these criteria, controlled processing should be allowed to modify and reautomate how we see ourselves and what policies we pursue.

A cognitive model of depression should explain the *cognitive loop*, the sequence of failure-related memories and thoughts consuming the depressed person's attention. This cognitive loop has been explained in terms of the network structure of episodic and emotional memory (Bower, 1981; Ingram, 1984), and the evoking of emotionally related memories (Blaney, 1986). Memories acquired during a particular emotional state become attached to emotion "nodes". Experiencing an emotion activates memories associated with the emotion, leading to "mood congruent" memory retrieval. A failure produces the subjective experience of being reminded of a series of unpleasant memories involving similar failures. Consciously experiencing each unpleasant memory evokes another mood congruent memory and the cycle becomes self-perpetuating.

What is the significance of the cognitive loop? One answer may lie in the role of repetition in automation. While controlled processes may suggest changes in the self-schema, for these changes to become automatic the self-schema must be trained against a database of relevant examples. Given a self-schema whose structure is based on pragmatic and emotional knowledge, mood congruent memory retrieval may serve as an index into memories of previous events most similar to the personal failure triggering the depression. The cognitive loop occurs because of the repetitive nature of "tuning" the self-schema against a database of test cases.

Support for the view that reactive depression involves a malleable self-schema can be found in the phenomenon of *slot rattling*. When asked to assess themselves in terms of a list of adjectives, the mildly depressed individual selects more negative adjectives on average, however day-to-day fluctuations occur (Ross, 1985). We would reasonably expect this variation if these adjectives reflected changing generalizations about ourselves.

A model of depression as the inductive restructuring of the self-schema partially explains an interesting observation. According to a major theory of depressive cognition, negative schemata distort experience to support a negative view of the depressed individual, the situation, and the future (Beck, 1979). While this theory is formulated with respect to clinical depression, it is intriguing that mildly depressed people are actually more *objective* about their performance at a variety of tasks than nondepressed people (Alloy & Abramson, 1979; Coyne & Gotlib, 1983; Giles & Shaw, 1987). This may be explained by hypothesizing that mildly depressed people pay more attention to information relevant to the modification and reautomation of the self-schema. Objective assessments about how well a task is performed are part of the multitude of observations used to inductively reconstruct the self-schema. Metaphorically, the self-schema is a hierarchical edifice resting on a foundation of observations of individual interactions with the world. Like the construction of a scientific theory, its validity rests on the objectivity of individual experimental observations.

A ubiquitous phenomena in depression is *loss of motivation*. In our model, this loss of motivation corresponds to the loss of automaticity of the descent of the self-schema. Automatic emotional responses to a situation cease to automatically and effortlessly formulate appropriate goals and plans. They require the effortful allocation of attention. We can conceive of motivation as a kind of "volunteerism" on the part of subportions of the self-schema. High in the hierarchy, a node might correspond to "getting up in the morning", a prerequisite for all subsequent activities of the day. Lower in the hierarchy a node might correspond to "go to work". Normally we get out of bed, go to work, and start conversations "automatically". During depression these responses are no longer automatic, and directing attention to appropriate goals and plans becomes *effortful*. Descending the self-schema requires allocation of attention away from other tasks. In the mild periodic depression that we are theorizing about, this may be more of a feature than a flaw. We should not be motivated to perform tasks using an attention-directing system that has recently failed, is undergoing modification, and may be preoccupied with a stream of failure-related thoughts and memories.

Depression Emulation Program

DEP is a computer simulation consistent with two sets of constraints. It must reflect the

central hypothesis that reactive depression is a form of failure-triggered controlled processing of the self-schema. It must also exhibit a set of empirically observed depressive behaviors. We will describe DEP's architecture while illustrating its gross behavior.

DEP has two subprograms: SS, for Self-Schema, and PEAM, for Pragmatic-Emotional Associative Memory. SS is a connectionist discrimination network with environmental inputs (Figure 2, SS). The weights connecting SS's nodes with each other and the environment represent the potentials for emotional reactions elicited by inputs from the environment. A descending, activated path to a leaf node is analogous to an emotional state (for example, the dark path in Figure 2), and the associated goals, plans, and expectations are accessed by activating the leaf node. Descent is either "automatic" or "controlled", depending on whether DEP is in a "nondepressed" or "depressed" state. The implementation of DEP's automatic processing is based on insights afforded by a sophisticated, and biologically plausible, computer simulation of automatic and controlled processing (Schneider, 1987) and will be discussed when we describe DEP's analog of decreased motivation in depression. DEP can adapt to a changing environment by changing the weights connecting SS's nodes to each other and the environment, which is analogous to learning new emotional responses for directing attention.

PEAM is an associative network linking SS leaf nodes to SS inputs (Figure 2, PEAM). Using a simple learning rule, PEAM's weights are continuously adjusted until it can predict the environmental inputs that produce particular SS responses. Given an SS output, PEAM can reproduce the inputs to SS that should cause it. This allows PEAM to be used to retrain SS's responses using a model of the environment, instead of the environment itself. Retraining a self-schema with a model of the environment has two advantages. The model can select more relevant training inputs, and learning with a model is less dangerous than learning with the real thing. PEAM's associative structure can be used to trigger SS controlled processing and then to provide inputs most effective for modifying SS (discussed below).

Figure 3 displays six statistics calculated with respect to a DEP simulation run. A training epoch is one exposure to each possible environmental input. SS and PEAM's weights are initially set to small random numbers and adjusted with simple learning rules until SS produces correct outputs to environmental inputs E_0 and PEAM produces correct inputs to SS given the SS outputs (Figure 3, E_0).

DEP's analog of a *failure* (Figure 3, (1)) is the generation of an incorrect response to environmental inputs. We can precipitate a failure by abruptly changing DEP's environment from E_0 to E_1 (Figure 3, "Environmental Change"). This is analogous to a personal failure caused by an automatic misdirection of attention to inappropriate goals and plans. However, a single failure should not induce DEP to modify and reautomate its attention-directing weights. The failure must meet the same criteria that predict vulnerability to depression in humans. It must be *stable*, *internal*, and *global* (Figure 3, (2)(3)(4)). The failure must be likely to reoccur (stable failure). The failure must be avoidable by changing DEP's self-schema

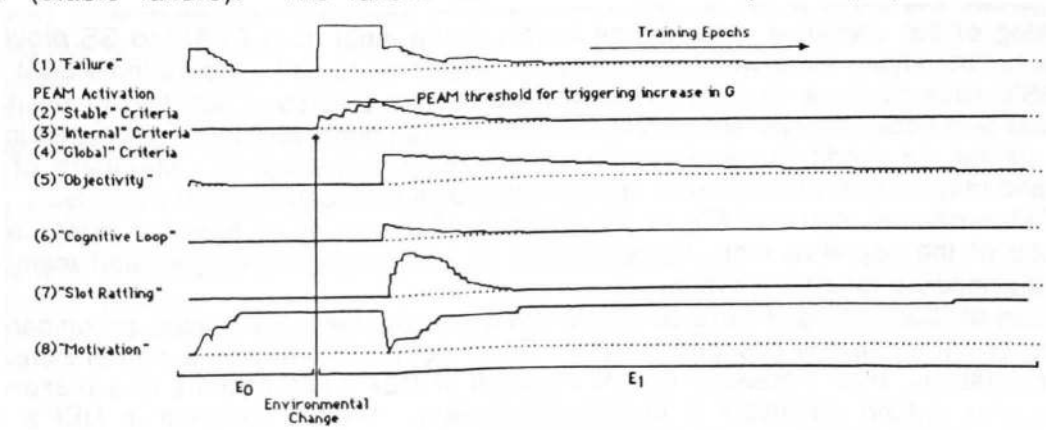


Figure 3: Output from DEP Simulation

(internal failure). The cause of the failure must potentially affect many different leaf nodes (global failure). All three of these criteria for triggering DEP's reactive depression can be achieved by using total PEAM activation to trigger controlled processing of DEP's self-schema.

PEAM accepts inputs and uses an interactive activation and competition paradigm (McClelland, 1981) to settle into an equilibrium state. PEAM's total activation depends on how much input it receives, where it receives the input, and how strongly individual nodes activate or suppress each other. Each time SS encounters a set of inputs and activates a leaf node, PEAM is activated with an input corresponding to SS's response. When SS is wrong and can find a correct response, the PEAM node associated with the correct response is also activated. If SS is wrong once (a non-stable failure), PEAM will not achieve a large activation because PEAM's weights have not had time to converge to a new model of the environment. If SS cannot find a correct response (a non-internal failure), PEAM will not achieve a large activation because PEAM does not receive the second input. If the two inputs (Figure 2, "False Negative Path" and "False Positive Path") are too similar (a non-global failure), PEAM will not achieve a large activation because not enough nodes are activated. Thus we can "gate" controlled processing of DEP's self-schema according to the criteria that a failure must be stable, internal, and global by establishing a threshold activation for DEP's analog of emotional memory. When PEAM activation (Figure 3, (2)(3)(4)) reaches threshold, DEP's depression commences. PEAM triggers DEP's depression by increasing a parameter (G) that increases the sensitivity of SS's weights to feedback from the environment about SS's performance.

DEP's analog for degree of *objectivity* (Figure 3, (5)) during reactive depression is the effect of a learning signal used to change SS weights based on the difference between SS's actual and desired output (a delta rule). Since increased sensitivity to differences between actual and desired performance is one kind of objectivity, increasing the effect of the learning signal has the effect of increasing DEP's analog of objectivity in the service of accurate self-schema modification. Activation of SS's nodes uses a standard logistic function (Rumelhart & McClelland, 1986), containing an additional parameter, G, divided into the net input and controlling the slope of the sigmoidal output of the logistic function. A small G results in extremely high or low SS node activation, and SS weights that are insensitive to the learning signal. A large G results in SS node activations near zero, and SS weights that are very sensitive to the learning signal. Decreasing G results in automatic descent of DEP's self-schema because all SS node activations are larger than a minimum threshold. Increasing G results in controlled descent because node activations fall below minimum thresholds and SS nodes must be sequentially searched to find the correct descent path. A small G is necessary for highly automated responses to a relatively static environment. A large G is necessary for learning new weight configurations in a changing environment. DEP's analog of emotional memory, PEAM, triggers modification and reautomation of DEP's self-schema by increasing G and allowing it to decay to zero. However, PEAM will only trigger controlled processing if the failure is stable, internal, and global, the same predictors for vulnerability to depression in humans.

DEP's analog of the *cognitive loop* (Figure 3, (6)) is the input from PEAM to SS providing training cases for modifying SS's weights (Figure 2, "Cognitive Loop"). Activating PEAM with inputs from SS's false positive and false negative leaf nodes creates a coalition of positively activated, similar leaf nodes. These are exactly the leaf nodes (along with their associated inputs to SS) most useful for modifying and reautomating DEP's self-schema. Members of this coalition are randomly selected to generate appropriate inputs and learning signals for SS . This sequence of failure-related inputs to SS, in the absence of actual inputs from the environment, is DEP's analog of the cognitive loop, the sequence of failure-related thoughts and memories preoccupying the mildly depressed person.

DEP's analog of *slot rattling* (Figure 3, (7)) is the increased weight variation accompanying a large learning signal and easily changed weights. Since slot rattling may be a sign of increased self-schema malleability, then increased weight variation in DEP's self-schema is a reasonable analog for the slot rattling observed in human depression. Weight variation in DEP's self-schema is especially severe if the environment is noisy and inconsistently rewards and punishes the same response.

DEP's analog of *lack of motivation* (Figure 3, (8)) is SS's lack of automaticity. Assuming "motivation" is how forcefully and persistently attention is directed to particular goals, motivation is how automatically goals are pursued. If an SS node (Figure 2, branch points in "Self-Schema") achieves activation above a minimum threshold, then it "automatically" transmits this activation, in parallel, to its daughter nodes at the next level down. If an activation falls below the minimum threshold, the node must wait for an external "controlling" signal to transmit. Increasing G increases feedback from the environment at the same time it causes all activations to move toward zero. As activations drop below minimum thresholds, more and more external controlling signals are necessary to achieve descent down DEP's self-schema. The external controlling signals (presumably requiring an "effortful" allocation of attention) are necessary to affect a sequential search for a correct descent path. As G decays back to zero, activations increase and descent of DEP's self-schema becomes more automatic, or "motivated". Thus, DEP must pass through a phase of controlled processing to learn new automatic processes. In order for DEP to change from one set of motivated, but incorrect, behaviors to a new set of equally motivated, and correct, behaviors, it must pass through a phase of unmotivated, but objective, modification and reautomation of its analog of the self-schema.

Conclusion

DEP is a highly simplified implementation of one way that emotions might normally direct attention to appropriate goals and plans, and periodically reorganize in the face of a changing environment. DEP exhibits a number of depressive information processing phenomena including: stable, internal, and global failures triggering qualitative changes in behavior, increased objectivity, the cognitive loop, fluctuations of self-describing adjectives, and decreased motivation. An emotional attention-directing system, grossly equated with the self-schema, normally operates in an automatic and motivated fashion insulated from environmental feedback. An emotional memory system constructs a model of the environment and periodically triggers reorganization of the emotional attention-directing system. Stable, internal, and global failures trigger the reorganization. These criteria follow from the self-schema's hierarchical structure, and are implemented using associative emotional memory. Once self-schema reorganization begins, emotional memory serves as a reservoir of training inputs to the self-schema. Loss of automaticity for engaging the environment causes a withdrawal that protects the vulnerable system while it reorganizes.

This computational model of reactive depression is one of many possible models. The DEP computer simulation forces us to examine our assumptions about what constitutes our theory. We need a more complete elaboration about what DEP's nodes and weights represent in terms of specific environmental stimuli, emotional processes, plans, and goals. One avenue is to represent a depressive scenario based on a case description of reactive depression. Another possibility is to relate DEP's parameters and behavior to some of the many standardized tests for diagnosing depression. Eventually it may be possible to relate DEP subsystems and processes to neurologic changes during depression. A rapidly evolving literature on the "neurology of depression" (Otto et al., 1987; Coffey, 1987) implicates the right cerebral hemisphere in depression. The right hemisphere plays a special, if not dominant, role in emotional, attentional, and pragmatic processes (Code, 1987; Kosslyn, 1987). Since our theory of reactive depression is about these topics, our computational model may provide a unique perspective to analyze previously disparate observations about the role of the right hemisphere in depression. Eventually we may be able to advance specific hypotheses about differences between mild-reactive and chronic-suicidal depression by understanding the differences in terms of perturbations in a cognitive model of reactive depression (Webster et al., 1988).

References

Alloy, L., & Abramson, L. (1979). Judgement of contingency in depressed and nondepressed students: Sadder but wiser. *Journal of experimental psychology*: General, 108, 441-485.

Beck, A., Rush, A., Shaw, B., & Emery G. (1979). *Cognitive therapy of depression*. New York: Guilford Press.

Blaney, P. (1986). Affect and memory: A review. *Psychological bulletin*, 99, 229-246.

Bower, G. (1981). Mood and memory. *American psychologist*, 36, 129-148.

Carver, C. & Scheier, M. (1986). Functional and dysfunctional responses to anxiety: The interaction between expectancies and self-focused attention. In R. Schwarzer (Ed.) *Self-related cognitions in anxiety and motivation*. Hillsdale, Lawrence Erlbaum.

Code, C. (1987). *Language, aphasia, and the right hemisphere*. New York: John Wiley & Sons.

Coffey, C. (1987). Cerebral laterality and emotion: The neurology of depression. *Comprehensive psychiatry*, 28, 197-219.

Coyne, J., & Gotlib, I. (1983). The role of cognition in depression: A critical appraisal. *Psychological Bulletin*, Vol. 94, 3, 472-505.

de Sousa, R. (1987). *The rationality of emotion*. Cambridge, MIT Press.

Dyer, M. (1983). *In-depth understanding*. Cambridge: MIT Press.

Feigenbaum, E. & Simon, H. (1984). EPAM-like models of recognition and learning. *Cognitive Science*, 8, 305-336.

Giles, D., & Shaw, B. (1987). Beck's cognitive theory of depression: Convergence of constructs. *Comprehensive psychiatry*, 28, 416-427.

Ingram, R. (1984). Toward an information-processing analysis of depression. *Cognitive therapy and research*, 8, 443-478.

Klerman, G. (1987). The nature of depression: mood, symptom, disorder. In A. Marsella, R. Hirschfeld, & M. Katz (Eds.), *The measurement of depression*. New York: Guilford Press.

Kosslyn, S. (1987). Seeing and imagining in the cerebral hemispheres: A computational approach. *Psychological review*, 94, 148-175.

Marcus, H. (1980). The self in thought and memory. In D. Wegner and R. Vallacher (Eds.), *The self in social psychology*. New York: Oxford University Press.

McClelland, J. (1981). Retrieving general and specific knowledge from stored knowledge of specifics. *Proc. of the third annual meeting of the cognitive science society*, 170-172.

Otto, M., Yeo, R., & Dougher, M. (1987). Right hemisphere involvement in depression: Toward a neuropsychological theory of negative affective experiences. *Biological psychiatry*, 22, 1201-1215.

Ross, M. (1985). Depression, self-concept, and personal constructs. In F. Epting & A. Landfield (Eds.), *Anticipating personal construct psychology*. Lincoln: University of Nebraska Press.

Rumelhart, D., McClelland, J. (1986) *Parallel distributed processing, Explorations in the microstructure of cognition, Vol. 1: Foundations* . Cambridge: MIT Press.

Schneider, W., Dumais, S., & Shiffrin, R. (1984). Automatic and control processing and attention. In *Varieties of attention*. New York: Academic Press.

Schneider, W., & Shiffrin, R. (1977). Controlled and automatic human information processing: I. Detection, search, and attention. *Psychological review*, 84, 1-66.

Schneider, W. & Detweiler, M. (1987). A connectionist/control architecture for working memory. In G. Bower (Ed.), *The psychology of learning and motivation* (Vol. 21), New York: Academy Press.

Seligman, M. (1984). Causal explanations as a risk factor for depression: Theory and evidence. *Psychological review*, 91, 347-374.

Shavelson, R. & Marsh, H. (1986). On the structure of the self-concept. In R. Schwarzer (Ed.) *Self-related cognitions in anxiety and motivation*. Hillsdale, Lawrence Erlbaum.

Shiffrin, R., & Schneider, W. (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending, and a general theory. *Psychological review*, 84, 127-190.

Simon, H. (1967). Motivational and emotional controls of cognition. *Psychological review*, 74, 29-39.

Webster, C., Sherman, H., & Banks, G. (1988). Computational neurology: The perturbation of normal models of cognition. *Proceedings of the AAAI symposium on artificial intelligence in medicine*, Stanford Univeristy, Palo Alto, CA.

INVITED ADDRESS

Reasoning by rule or model ?

P.N. Johnson-Laird
MRC Applied Psychology Unit, Cambridge, England

INTRODUCTION

Human reasoning depends on three principal skills: comprehension, the generation of conclusions, and the evaluation of conclusions. The critical step in a deduction is the evaluation of a conclusion in order to ensure it is valid. Some theorists propose that this process is akin to a formal derivation in a logical calculus (see e.g. Inhelder and Piaget, 1958; Braine, 1978; Rips, 1983). Other theorists propose that it depends on a search for alternative interpretations, or *models*, of the premises that serve as refutations (e.g. Newell, 1981; Johnson-Laird, 1983). No evidence so far has been universally accepted as counting decisively against one school of thought or the other. My aim in this paper is to try to settle the issue by considering a class of inferences not hitherto investigated experimentally -- multiply-quantified deductions.

An example of a multiply-quantified assertion is: None of the artists is taller than any of the beekeepers. Such assertions contain a relational expression -- here, "taller than" -- and its arguments are quantified using such expressions as "all", "some", "none". These quantifiers behave in ways that are similar to those of the first-order predicate calculus (though there are others that do not). To explain how people reason with multiple quantifiers, it is necessary first to account for how they reason with relational expressions. I will describe a theory of relational reasoning and then some evidence from a crucial experiment carried out to compare this theory and a rule-based theory. Next, I will present a theory of reasoning with multiple quantifiers and evidence from a second experiment designed to decide between this theory and a rule-based theory. Finally, I will outline the essential theoretical difference between the two sorts of theory. All this work was carried out in collaboration with Ruth Byrne.

RELATIONAL REASONING

Most psychological studies of relational reasoning have concerned so-called three-term series problems, such as: Anne is taller than Betty; Carol is shorter than Betty; who is tallest? The evidence from this domain has not sufficed to decide between rule-based and model-based theories (see e.g. Huttenlocher, 1968; Clark, 1969). The validity of these inferences depends on the transitivity of the relation in the premises, and theorists have generally assumed that there is either a general schema for transitivity or content-specific rules, such as:

if x is taller than y and y is taller than z, then x is taller than z.

JOHNSON-LAIRD

The theory based on the manipulation of mental models proposes that the logical properties of relations, such as transitivity, are not explicitly represented at all, but are emergent from the *meanings* of relations (Johnson-Laird, 1983). The meaning of an assertion such as:

The number of artists is greater than the number of beekeepers

enables the interpretative system to construct a model of the situation, e.g.

artist	beekeeper
artist	beekeeper
artist	beekeeper
artist	

and to verify that the relation holds within models. The meaning of "greater than" can be formulated in terms of the concepts of an empty set, a non-empty set, and the addition and subtraction of items from sets:

The number of x's is greater than the number of y's: if there is at least one x and there are no y's or (x - 1) is greater than (y - 1).

There are other ways of giving a recursive definition of the concept, and such definitions can be couched in a form that is suitable for the construction and manipulation of models such as the one above. Hence, given the further assertion:

The number of beekeepers is greater than the number of chemists

the interpretative system can construct the model:

artist	beekeeper	chemist
artist	beekeeper	chemist
artist	beekeeper	
artist		

which supports the conclusion that the number of artists is greater than the number of chemists. To test validity, it is necessary to search for an alternative model of the premises that refutes the conclusion. There are various ways in which the search could be made, and some have been modelled computationally (Johnson-Laird and Bara, 1984). But, here I will not make any strong claims about the procedures that people use, other than that they do not possess any simple deterministic algorithm for searching for refutations (cf. Newell and Simon, 1972). Hence, where a correct response can be made only as a result of considering more than one model, the theory predicts that the task will be reliably harder -- a prediction that has been confirmed in many studies of traditional syllogisms, i.e. arguments that depend on singly-quantified premises (see e.g.

Johnson-Laird and Bara, 1984). We can predict that when people reason about relations, the task should be harder where there is a genuine choice of models of the same premises.

Ruth Byrne and I tested this prediction by examining such descriptions as:

The jug is on the right of the cup
The plate is on the left of the cup
The knife is in front of the plate
The fork is in front of the jug.

The description corresponds to a single determinate model:

plate	cup	jug
knife		fork

and so it should be relatively easy to answer a question about the relation between the knife and the fork: The knife is on the left of the fork. But, when the second premise is instead:

The plate is on the left of the jug

the description is consistent with at least two distinct models:

plate	cup	jug	cup	plate	jug
knife		fork	knife	fork	

The same relation holds between the knife and the fork in either layout, but the model-based theory predicts that the task should be harder because both models ought to be constructed in testing the validity of the answer. The task should be still harder where the correct response can be made only by constructing both models. The following description:

The jug is on the right of the cup
The plate is on the left of the jug
The knife is in front of the plate
The fork is in front of the cup

is consistent with two distinct models:

plate	cup	jug	cup	plate	jug
knife	fork		fork	knife	

that have no relation in common between the knife and fork, and so there is no valid conclusion. Granted that working memory has a limited processing capacity, the model-based theory predicts

the following rank order of increasing difficulty: one-model problems, multiple-model problems with valid conclusions, and multiple-model problems with no valid conclusions.

A formal rule for making such inferences needs to be of the following sort: If x is related to y on one dimension, and w is related to x on an orthogonal dimension, and z has the same orthogonal relation to y, then w is related to z in the same way as x is related to y. The rule can be applied to the configuration:

```
x ------- y
|         |
w         z
```

in any orientation. Moreover, it can be directly applied to the premises yielding a multiple-model problem with a valid conclusion (see above) because the relation between x and y is directly asserted by the premise: The plate is on the left of the jug. But there is no such premise in the one-model problem: the relevant relation has to be inferred (using a further rule of inference) before the present rule can be applied. Hence, the rule-based theory predicts that one-model problems should be harder than multiple-model problems with valid concludions -- exactly the opposite prediction to the model-based one.

We have carried out a series of experiments in order to compare the predictions of the two theories. In our latest experiment, 18 adult subjects acted as their own controls and carried out four inferences of each of the three sorts. The percentages of their correct responses were: 70% for one-model problems, 46% for the multiple-model problems with valid conclusions, and 8% for the multiple-model problems with no valid conclusion. This reliable trend corroborates the model-based theory but runs counter to the theory based on rules.

MULTIPLY-QUANTIFIED REASONING

Granted the following definition of simple consanguineal relationships:

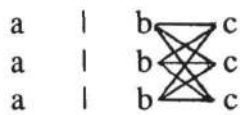
> x is related to y if x is a parent of y (or x is a child of y) or x is a parent of z (or x is a child of z) and z is related to y

it is an emergent property that the relation is symmetric and transitive. Thus, the following multiply-quantified premises:

> None of the artists is related to any of the beekeepers.
> All of the beekeepers are related to all of the chemists.

yield the valid conclusion: None of the artists is related to any of the chemists.

The model-based theory assumes that reasoners construct a model of the state of affairs described by the premises:

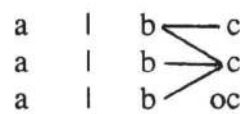


where the vertical barrier represents negation and prevents any relation between the artists and beekeepers, and each link corresponds to a relation between two individuals. It is impossible to construct any model of the premises that refutes the conclusion.

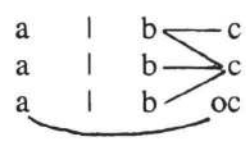
The following premises, which differ only in a single quantifier:

> None of the artists is related to any of the beekeepers.
> All of the beekeepers are related to some of the chemists.

yield a similar model:



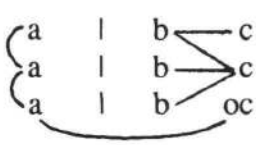
from which the same conclusion as before can be drawn. But in this case a search for a counterexample will be successful:



The two models support only the conclusion:

> Some of the artists are not related to any of the chemists.

but again this conclusion can be refuted by a further model:



The only valid conclusions are:

> Some of the chemists are not related to any of the artists.
> None of the artists is related to *some* of the chemists.

or weaker ones, e.g. the ambiguous assertion, None of the artists are related to all of the chemists. Plainly, the model-based theory predicts that the one-model inference should be easier than the multiple-model inference.

Although there are rule-based theories of reasoning with quantifiers (e.g. Braine and Rumain, 1983), there is no current theory that is powerful enough to derive multiply-quantified conclusions. However, the two sorts of inference can be derived in the first-order predicate calculus. Both proofs are long -- at least fifteen lines of derivation -- but they are remarkably similar: they require the same number of steps, and the rules that are used at each step are identical except in two cases, and even here there are only minor differences. Since the number of steps in a derivation is taken to predict the psychological difficulty of an inference (see e.g. Rips, 1983), the rule-based theory does not predict any difference between the two sorts of problem.

Ruth Byrne, Patrizia Tabossi, and I have carried out a series of experiments to investigate multiply-quantified reasoning in which the subjects drew their own conclusions in their own words. In one of these experiments, we tested 11 adults with a series of problems including the two described above and other similar ones. They made 77% correct responses for the one-model problems, but only 23% correct responses for the multiple-model problems -- a difference that was highly significant. The only divergence between the rule-based derivations of the two sorts of problem concerns the elimination and subsequent re-introduction of quantifiers: the one-model problem has these steps for the quantifier "all" where the multiple-model problem has them for "some". Could there be an intrinsic difference in the difficulty of the two classes of rules? Another sort of one-model problem in the experiment called for the use of the rules for "some", e.g.: Some of the beekeepers are related to all the artists; None of the chemists are related to any of the beekeepers. This problem was relatively easy with 67% correct conclusions. Hence, there is no intrinsic difficulty in dealing with "some".

CONCLUSIONS

The model-based theory extends naturally to reasoning with premises containing multiply-quantified relations. Rule-based theories, however, appear neither to predict the relative difficulty of inferences or the typical errors that reasoners make. A fundamental difference between the two approaches concerns the treatment of variables. Theories based on formal rules propose that reasoners, having gone to the trouble of understanding the premises, base their inferences on something other than a full semantic interpretation. They are supposed to abstract the underlying logical skeleton -- so-called "logical form" -- and then exploit formal rules of instantiation in order to replace quantified variables by *single* hypothetical individuals that stand in for them; after a stage of formal reasoning about sentential connectives, quantifiers can then be restored by formal rules of generalization operating on these hypothetical individuals. According to the model-based theory, however, the work of instantiation is merely part of the normal process of comprehension: universally quantified phrases are instantiated by sets of mental tokens that are treated as exhaustively representing the relevant set; existential quantified phrases are similarly instantiated

by sets of mental tokens, except that there are optional items that do not satisfy the conditions of the assertion. One of the consequences of this semantic distinction is that the choice of quantifier in one premise can affect the number of possible models of the premises as a whole.

Of course it may be possible to devise a theory based on formal rules that will account for our experimental results -- rule-based theories in general have universal Turing machine power. Nevertheless, the lack of any currently feasible rule-based theory may not be accidental. The theory of mental models is, by comparison, relatively simple to refute. It predicts that whenever the meaning of premises supports more than one conclusion about what is possible though not necessary, there are multiple models of the premises, and so the inferential task will be harder. This prediction has now withstood empirical testing for three domains of inference: syllogisms, relational reasoning, and reasoning with multiple quantifiers.

REFERENCES

Braine, M.D.S. (1978) On the relation between the natural logic of reasoning and standard logic. *Psychological Review*, 85, 1-21.

Braine, M.D.S., and Rumain, B. (1983) Logical reasoning. In Flavell, J.H., and Markman, E.M. (Eds.) *Handbook of Child Psychology, Vol. III, Cognitive Development.* 4th Ed. New York: Wiley.

Clark, H.H. (1969) Linguistic processes in deductive reasoning. *Psychological Review*, 76, 387-404.

Huttenlocher, J. (1968) Constructing spatial images: A strategy in reasoning. *Psychological Review*, 75, 550-560.

Inhelder, B., and Piaget, J. (1958) *The Growth of Logical Thinking from Childhood to Adolescence.* London: Routledge & Kegan Paul.

Johnson-Laird, P.N. (1983) *Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness.* Cambridge: Cambridge University Press. Cambridge, MA: Harvard University Press.

Johnson-Laird, P.N., and Bara, B.G. (1984) Syllogistic inference. *Cognition*, 16, 1-61.

Newell, A. (1981) Reasoning, problem solving and decision processes: the problem space as a fundamental category. In Nickerson, R. (Ed.), *Attention and Performance, Vol. 8.* Hillsdale, NJ: Erlbaum.

Newell, A. and Simon, H. (1972) *Human Problem Solving.* Englewood Cliffs, NJ: Prentice-Hall.

Rips, L.J. (1983) Cognitive processes in propositional reasoning. *Psychological Review*, 90, 38-71.

NEW AND FORTHCOMING TITLES FROM ABLEX

HUMAN / COMPUTER INTERFACE DESIGN GUIDELINES
C. Marlin Brown

HUMAN FACTORS IN MANAGEMENT INFORMATION SYSTEMS
Jane Carey, ed.

ADVANCES IN HUMAN / COMPUTER INTERACTION VOL. 2
H. Rex Hartson & Deborah Hix, eds.

COGNITION AND SYMBOLIC STRUCTURES
Robert Haskell, ed.

EXPERT SYSTEMS: THE USER INTERFACE
James A. Hendler, ed.

ONLINE HELPS: DESIGN AND IMPLEMENTATION
Greg Kearsley

FORMALIZING COMMON SENSE
Vladimir Lifschitz & John McCarthy

EMPIRICAL STUDIES OF PROGRAMMERS VOL. 2
Gary Olson, Elliot Soloway, & Sylvia Sheppard, eds.

COMPUTATIONAL PROCESSES IN HUMAN VISION
Zenon W. Pylyshyn, ed.

PERSPECTIVES ON THE COMPUTER REVOLUTION
Zenon W. Pylyshyn & Liam Bannon, eds.

AI TOOLS AND TECHNIQUES
Mark Richer, ed.

COGNITION, COMPUTING, AND COOPERATION
Scott P. Robertson, Wayne Zachary & John B. Black, eds.

MODELS OF COGNITION: A REVIEW OF COGNITIVE SCIENCE
Noel E. Sharkey, ed.

FROM SYNTAX TO SEMANTICS: INSIGHTS FROM MACHINE TRANSLATION
E.H. Steiner, P. Schmidt & C. Zelinsky-Wibbelt, eds.

IMAGE UNDERSTANDING 1987 / 88
Shimon Ullman & Whitman Richards, eds.

CONNECTIONIST MODELS AND THEIR IMPLICATIONS
David Waltz & Jerome Feldman, eds.

REQUIREMENTS FOR A SOFTWARE ENGINEERING ENVIRONMENT
Marvin Zelkowitz, ed.

ADVANCES IN NATURAL LANGUAGE GENERATION VOLS. 1-2
Michael Zock & Gerard Sabah, eds.

ABLEX JOURNALS
- COGNITIVE SCIENCE
- DISCOURSE PROCESSES
- NEW INTERNATIONAL JOURNAL OF COMPUTER-AIDED VLSI DESIGN
- NEW JOURNAL OF COMPUTING AND SOCIETY

These and other backlist titles will be on display at the Ablex booth.
A 15% convention discount will apply to all book orders.

Ablex Publishing Corporation • 355 Chestnut St., Norwood, NJ 07648 • (201) 767-8450

For These Titles And More... Visit Our Booth

DISCOVERING CAUSAL STRUCTURE
Artificial Intelligence, Philosophy of Science, and Statistical Modeling
Clark Glymour, Richard Scheines, Peter Spirtes, and Kevin Kelly
with a Foreword by
Herbert A. Simon
Casebound edition including 9-variable program on floppy disk: $39.50/ISBN: 0-12-286961-3
1987, 394 pages
23-variable program available separately: $200.00/ISBN: 0-12-286962-1

THE STRUCTURE OF WRITTEN COMMUNICATION
Studies in Reciprocity between Writers and Readers
Martin Nystrand
1986, 208 pages, $29.95
ISBN: 0-12-523482-1

COMPREHENDING ORAL AND WRITTEN LANGUAGE
edited by
Rosalind Horowitz and S. Jay Samuels
1987, 430 pages, $42.00
ISBN: 0-12-356110-8

THE DEVELOPMENT OF SCIENTIFIC THINKING SKILLS
Deanna Kuhn, Eric Amsel, and Michael O'Loughlin
with the assistance of
Leona Schauble, Bonnie Leadbeater, and William Yotive
1988, 249 pages, $29.95
ISBN: 0-12-428430-2

TRANSFER OF LEARNING
Contemporary Research and Applications
edited by
Stephen M. Cormier and Joseph D. Hagman
A Volume in the EDUCATIONAL TECHNOLOGY Series
1987, 281 pages, $39.50
ISBN: 0-12-188950-5

HUMAN FACTORS IN AVIATION
edited by
Earl L. Wiener and David C. Nagel
1988, 556 pages, $65.00
ISBN: 0-12-750030-8

PSYCHOLOGY OF LEARNING AND MOTIVATION
Advances in Research and Theory
Volume 21
edited by
Gordon H. Bower
1988, 328 pages, $55.00
ISBN: 0-12-543321-2

LANGUAGE PERCEPTION AND PRODUCTION
Relationships between Listening, Speaking, Reading, and Writing
edited by
Alan Allport, Donald G. MacKay, Wolfgang Prinz, and Eckart Scheerer
1987, 497 pages, $64.00
ISBN: 0-12-052750-2

NEW DIRECTIONS IN SEMANTICS
edited by
Ernest Le Pore
A Volume in the COGNITIVE SCIENCE Series
In Paperback: $39.00
ISBN: 0-12-444041-X
Casebound: $76.00
ISBN: 0-12-444040-1
1987, 400 pages

ACADEMIC PRESS
Harcourt Brace Jovanovich, Publishers
Book Marketing Dept. #07088, 1250 Sixth Ave., San Diego, CA 92101
Call Toll Free 1-800-321-5068

© 1988 by Academic Press, Inc. All Rights Reserved.

Prices subject to change without notice. CB/SJ—07088

The Latest in Cognitive Science
From Bradford Books

NEUROCOMPUTING
Foundations of Research
edited by James A. Anderson and Edward Rosenfeld

A fundamental reference book that collects seminal work by McCulloch and Pitts, Hebb, Lashley, von Neumann, Minsky and Papert, Cooper, Grossberg, Kohonen, and McClelland and Rumelhart.
June 8½ x 11 800 pp. 01097-6 $55.00

NEURAL NETWORKS AND NATURAL INTELLIGENCE
Stephen Grossberg

This anthology of the latest research in neural networks is packed with real-time computer simulations and rigorous demonstrations, covering results in vision, speech, cognitive information processing, adaptive pattern recognition, adaptive robotics, conditioning and attention, cognitive-emotional interactions, and decision making under risk.
June 7 x 10 544 pp. 07107-X $35.00

BELIEF IN PSYCHOLOGY
A Study in the Ontology of Mind
Jay L. Garfield

How should cognitive science treat the propositional attitudes — states such as beliefs, desires, hopes and fears? Jay Garfield asserts that the propositional attitudes can and must play useful theoretical roles in the science of the mind and stresses the importance of their social context in this sophisticated and original argument.
Available 6 x 9 208 pp. 07108-8 $19.95

EXPLAINING BEHAVIOR
Reasons in a World of Causes
Fred Dretske

Providing a direct response to recent arguments claiming that what we believe has no genuine explanatory power, Dretske develops a theory of what behavior is (contrasting it to simple bodily movements) and shows how behavior, so understood, can be causally explained by an agent's reason.
June 6 x 9 184 pp. 04094-8 $22.50

INTENSIONAL LOGIC AND THE METAPHYSICS OF INTENTIONALITY
Edward N. Zalta

Zalta tackles the issues that arise in connection with intensional logic — a formal system for representing and explaining the apparent failures of certain important principles of inference — and intentional states — mental states such as beliefs, hopes, and desires.
June 6 x 9 256 pp. 24027-0 $27.50

COGNITIVE SCIENCE IN MEDICINE
Biomedical Modeling
edited by David A. Evans and Vimla L. Patel

Cognitive Science in Medicine presents current research that focuses on issues and results in applying techniques from cognitive science to problems in biomedicine.
August 6 x 9 350 pp. 05037-4 $30.00

REPRESENTATION AND REALITY
Hilary Putnam

"With striking candor, Putnam exposes the factors that have shaped his thinking about intentionality. Since that thinking has had a great influence, this book is full of valuable insights into current philosophical methods, foibles, and aspirations."
— Daniel Dennett, Tufts University
July 6 x 9 160 pp. 16108-7 $19.95

KNOWLEDGE IN FLUX
Modeling the Dynamics of Epistemic States
Peter Gärdenfors

Knowledge in Flux presents a theory of rational changes of belief, focusing particularly on revisions that occur when the agent receives new information that is inconsistent with the present epistemic state.
August 6 x 9 224 pp. 07109-6 $27.50

MATTER AND CONSCIOUSNESS
Revised Edition
Paul M. Churchland

"Churchland's *Matter and Consciousness* remains a compact treasure trove for anyone interested in an instant overview of the mind-body problem, related philosophical issues, and their connection with the developing neuro- and psychological sciences of our day." — Keith Gunderson, *Philosophy of Science*
Available 6 x 9 208 pp. 03135-3 $20.00

For these and other exciting new books, visit us at our table at the Cognitive Science Society meeting.

ISBN Prefix: 0-262

The MIT Press
55 Hayward Street Cambridge, MA 02142

MORGAN KAUFMANN PUBLISHERS

READINGS IN COGNITIVE SCIENCE: *A Perspective from Psychology and Artificial Intelligence*

Edited by Allan Collins (Bolt Beranek and Newman) and Edward E. Smith (University of Michigan)

A collection of the best papers in cognitive science from the perspective of two of its contributing disciplines. The editors have selected key works on the central research topics: representation, categorization, learning, thinking, and perception. An opening section collects classic papers on the foundational issues in cognitive science. Introductions to each major section relate research to issues. Accessible to a broad audience of researchers and students in cognitive science, AI, psychology, cognitive pyschology, and related areas.

Fall 1988; 630 pages; paper; ISBN 0-55860-013-2; prepublication price $28.95

LEXICAL AMBIGUITY RESOLUTION: *Perspectives from Psycholinguistics, Neuropsychology, and Artificial Intelligence*

Edited by Steven L. Small (University of Rochester), Garrison W. Cottrell (University of California, San Diego), and Michael K. Tanenhaus (University of Rochester)

This book focuses on a core problem in the processing of human language. Eighteen original papers offer viewpoints and solutions from cognitive science, AI, psycholinguistics, neuropsychology, and theoretical linguistics. Part I, *Computer Models*, examines attempts to understand disambiguation through implementing computer programs. Part II, *Empirical Studies*, goes into the lab to examine the nature and structure of ambiguity itself. A valuable source work for researchers in language understanding and computational semantics.

1988; 518 pages; cloth; ISBN 0-934613-50-8; $34.95

ARTIFICIAL INTELLIGENCE AND TUTORING SYSTEMS: *Computational and Cognitive Approaches to the Communication of Knowledge*

Etienne Wenger (University of California, Irvine)

The first comprehensive reference text in this dynamic area provides a comprehensive survey of research, highlighting the field's major themes, and assessing the current state of the art. Adopting the perspective of the communication of knowledge, the author addresses practical issues involved in designing instructional systems, as well as theoretical questions raised by investigating computational methods of knowledge communication. *Artificial Intelligence and Tutoring Systems* can serve as an introduction for newcomers to the field and a valuable resource for researchers and practitioners.

1987; 486 pages; cloth; ISBN 0-934613-26-5; $34.95

PROCEEDINGS OF THE DARPA WORKSHOP ON CASE-BASED REASONING

Edited by Janet Kolodner (Georgia Institute of Technology)

Case-based reasoning focuses on techniques for making inferences from previous cases. This self-contained volume reprints classic background papers and reviews research-in-progress sponsored by DARPA. Theoretical topics are balanced by applications to the military, law, manufacturing, medicine, and instruction.

1988; 474 pages; paper; ISBN 0-934613-93-1; $30.00

NEW—VIDEOTAPED COURSES

The Linguistic Institute Videotape Series: The Stanford Video Journal comprises eleven full-length courses (8 to 22 lectures) in theoretical and computational linguistics offered at the 1987 Linguistic Institute at Stanford University. ***The Stanford Computer Science Video Journal: Artificial Intelligence Research Lectures*** presents taped individual lectures originally offered in Stanford University research seminars. Some survey major paradigms guiding AI research, including cognitive science, while others summarize results from specific projects. For complete information and prices, please check the box in the coupon below.

PROCEEDINGS OF THE FIFTH INTERNATIONAL CONFERENCE ON MACHINE LEARNING

Edited by John Laird (University of Michigan, Ann Arbor)

Original articles in machine learning—explanation-based and empirical methods, genetic algorithms, connectionist learning, probabilistic methods, and formal theories of learning. An up-to-the-minute record of current research in this growing field, of certain interest to researchers and students in AI, cognitive science, the philosophy of science, and education.

1988; 467 pages; paper; ISBN 0-934613-64-8; $24.95

ADVANCES IN NEURAL NETWORK INFORMATION PROCESSING SYSTEMS

Edited by David Touretzky (Carnegie Mellon)

This volume is an important compilation of research results from scientists in neurobiology, cognitive science, physics, and computer science and engineering. Based on papers presented at the second annual conference (November, 1988), major topics include: neurobiological models of learning and memory, connectionist models of cognitive processing; and applications in signal processing, vision, and speech, and simulation of neural networks. With index and extensive bibliographical material.

Spring 1989; approx 800 pages; cloth; ISBN 0-55860-015-9; prepublication price $33.95

NEW RESEARCH NOTES IN ARTIFICIAL INTELLIGENCE

Analogica

Edited by Armand Prieditis (Rutgers University)

A multidisciplinary collection of original papers presented by leading scientists at the first workshop on analogical reasoning, at Rutgers University, December, 1985.

1988; 176 pages; paper; ISBN 0-934613-37-0; $22.95

Semantic Networks: An Evidential Formalization And Its Connectionist Realization

Lokendra Shastri (University of Pennsylvania)

Shastri presents a massively parallel network of simple processing elements—inspired, in part, by the architecture of the animal brain.

1988; 222 pages; paper; ISBN 0-934613-39-7; $22.95

A Connectionist Approach To Word Sense Disambiguation

Garrison Cottrell (University of California, San Diego)

A new model for representing parsing of natural language, which accounts for recent psycholinguistic and neurolinguistic data. An inherently parallel approach suitable for implementation on a connection machine.

Fall 1988; 232 pages; paper; ISBN 0-934613-61-3; $22.95

Morgan Kaufmann Publishers, Inc.
Order Fulfillment Center, Dept. CO
P.O. Box 50490, Palo Alto, CA 94303

Please enclose full payment plus $2.25 for the first volume and $1.00 for each additional volume for book rate U.S. shipment. (For surface shipment outside the U.S.: $4.00 for the first volume and $3.00 for each additional volume.) California residents please add sales tax. If not satisfied, you may return books within 30 days for a full refund.

☐ Please send complete information on the videotapes

QTY.	ISBN#	First Author	Price
		Shipping	
		Tax	
		Total Enclosed	

Name
Address
City, State, Zip

NOW AVAILABLE

from LEA

PROCEEDINGS OF THE

ninth (1987)
eighth (1986)
seventh (1985)
sixth (1984)
fifth (1983)
fourth (1982)
third (1981)

ANNUAL CONFERENCES OF THE

COGNITIVE SCIENCE SOCIETY

$49.95 each, paperbound

Lawrence Erlbaum Associates, Inc, Publishers
365 Broadway, Hillsdale,NJ 07642